Dionex_COMPASS_May2022

Stephanie J. Wilson

2023-01-23

Daily Set up

Read in and Format the raw data - change wd & file names

```
#setwd("S:/Biogeochemistry/People/Wilson (Steph)/Data/Dionex/Raw Data Files")

# SULFATE DATA:

## Read in raw data file from Dionex - copied and saved as a txt
Sdat <- read.table("Raw Data/COMPASS_Synoptic_CB_MonMon_202205_SO4.txt",sep='\t' , header=T, skip=3)
head(Sdat)
```

```
##   X        X.1                  X.2 IC.SO4 IC.SO4.1 IC.SO4.2 IC.SO4.3 IC.SO4.4
## 1 1  Lab Blank              Unknown   n.a.     n.a.     n.a.     n.a.     n.a.
## 2 2  Lab Blank              Unknown   n.a.     n.a.     n.a.     n.a.     n.a.
## 3 3  Lab Blank              Unknown   n.a.     n.a.     n.a.     n.a.     n.a.
## 4 4  Lab Blank              Unknown   n.a.     n.a.     n.a.     n.a.     n.a.
## 5 5 Standard 1 Calibration Standard 4.133   0.4711     6.47   0.4192     2.16
## 6 6 Standard 2 Calibration Standard 4.130   0.9084     6.22   0.8083     4.14
```

```
## Only keep the columns that we need
Sdat <- Sdat[ ,c(2,5)] # dont need this here
head(Sdat)
```

```
##          X.1 IC.SO4.1
## 1  Lab Blank     n.a.
## 2  Lab Blank     n.a.
## 3  Lab Blank     n.a.
## 4  Lab Blank     n.a.
## 5 Standard 1  0.4711
## 6 Standard 2  0.9084
```

```
## Name the columns correctly
colnames(Sdat) <- c( "Sample_ID", "SO4_ppm")
Sdat$Sample_ID <- as.factor(Sdat$Sample_ID)
Sdat$SO4_ppm <- as.numeric(Sdat$SO4_ppm)
```

```
## Warning: NAs introduced by coercion
```

```
Sdat <- as.data.frame(Sdat)
head(Sdat)
```

```
##    Sample_ID SO4_ppm
## 1  Lab Blank      NA
## 2  Lab Blank      NA
## 3  Lab Blank      NA
## 4  Lab Blank      NA
## 5 Standard 1  0.4711
## 6 Standard 2  0.9084
```

```
#Chloride data
## Read in raw data file from Dionex - copied and saved as a txt
Cldat <- read.table("Raw Data/COMPASS_Synoptic_CB_MonMon_202205_Cl.txt",sep='\t' , header=T, skip=3)
head(Cldat)
```

```
##   X        X.1                  X.2  IC.Cl  IC.Cl.1 IC.Cl.2  IC.Cl.3 IC.Cl.4
## 1 1  Lab Blank              Unknown   n.a.     n.a.    n.a.     n.a.    n.a.
## 2 2  Lab Blank              Unknown   n.a.     n.a.    n.a.     n.a.    n.a.
## 3 3  Lab Blank              Unknown   n.a.     n.a.    n.a.     n.a.    n.a.
## 4 4  Lab Blank              Unknown   n.a.     n.a.    n.a.     n.a.    n.a.
## 5 5 Standard 1 Calibration Standard 3.243   5.0098   92.77   6.0066   38.67
## 6 6 Standard 2 Calibration Standard 3.243  10.0990   93.21  12.1084   77.34
##   IC.Cl.5
## 1    n.a.
## 2    n.a.
## 3    n.a.
## 4    n.a.
## 5     BMB
## 6     BMB
```

```
## Only keep the columns that we need
Cldat <- Cldat[ ,c(2,5)]
head(Cldat)
```

```
##          X.1  IC.Cl.1
## 1  Lab Blank     n.a.
## 2  Lab Blank     n.a.
## 3  Lab Blank     n.a.
## 4  Lab Blank     n.a.
## 5 Standard 1  5.0098
## 6 Standard 2 10.0990
```

```
## Name the columns correctly
colnames(Cldat) <- c( "Sample_ID", "Cl_ppm")
Cldat$Sample_ID <- as.factor(Cldat$Sample_ID)
Cldat$Cl_ppm <- as.numeric(Cldat$Cl_ppm)
```

```
## Warning: NAs introduced by coercion
```

```r
Cldat <- as.data.frame(Cldat)
head(Cldat)
```

```
##    Sample_ID  Cl_ppm
## 1  Lab Blank      NA
## 2  Lab Blank      NA
## 3  Lab Blank      NA
## 4  Lab Blank      NA
## 5 Standard 1  5.0098
## 6 Standard 2 10.0990
```

```r
## Bring the data back together:
all_dat <- merge(Sdat, Cldat, by="Sample_ID")
head(all_dat)
```

```
##   Sample_ID  SO4_ppm    Cl_ppm
## 1           1122.213 8909.0812
## 2           1122.213 3574.4197
## 3           1122.213    5.0098
## 4           1122.213 3462.5246
## 5           1122.213        NA
## 6            429.473 8909.0812
```

```r
## Remove empty lines
all_dat <- all_dat[!(is.na(all_dat$Sample_ID) | all_dat$Sample_ID==""), ]
head(all_dat)
```

```
##                              Sample_ID  SO4_ppm   Cl_ppm
## 26       COMPASS_20220502_MSM_SipA_10cm 924.7531 5530.741
## 27       COMPASS_20220502_MSM_SipA_20cm 686.1893 5431.449
## 28       COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549
## 29   COMPASS_20220502_MSM_SipA_45cm_Dup 398.1761 6525.085
## 30 COMPASS_20220502_MSM_SipA_45cm_Spike 606.2450 6339.224
## 31       COMPASS_20220502_MSM_SipB_10cm 747.5236 5169.818
```

```r
all_dat <- droplevels(all_dat[!all_dat$Sample_ID == 'Lab Blank',])
head(all_dat)
```

```
##                              Sample_ID  SO4_ppm   Cl_ppm
## 26       COMPASS_20220502_MSM_SipA_10cm 924.7531 5530.741
## 27       COMPASS_20220502_MSM_SipA_20cm 686.1893 5431.449
## 28       COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549
## 29   COMPASS_20220502_MSM_SipA_45cm_Dup 398.1761 6525.085
## 30 COMPASS_20220502_MSM_SipA_45cm_Spike 606.2450 6339.224
## 31       COMPASS_20220502_MSM_SipB_10cm 747.5236 5169.818
```

Pull out standards - could do some checks here if we want

```r
stds <- all_dat[grepl("Standard", all_dat$Sample_ID),]
head(stds)
```

```
##      Sample_ID SO4_ppm Cl_ppm
## 184 Standard 1  0.4979 5.2975
## 185 Standard 1  0.4979 5.0098
## 186 Standard 1  0.4979 5.1659
## 187 Standard 1  0.4979 5.2056
## 188 Standard 1  0.4711 5.2975
## 189 Standard 1  0.4711 5.0098
```

Calculate mmol/L concentrations

```r
#remove standards from sample dataframe
sampledat <- all_dat[!grepl("Standard", all_dat$Sample_ID),]
head(sampledat)
```

```
##                               Sample_ID  SO4_ppm   Cl_ppm
## 26       COMPASS_20220502_MSM_SipA_10cm 924.7531 5530.741
## 27       COMPASS_20220502_MSM_SipA_20cm 686.1893 5431.449
## 28       COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549
## 29   COMPASS_20220502_MSM_SipA_45cm_Dup 398.1761 6525.085
## 30 COMPASS_20220502_MSM_SipA_45cm_Spike 606.2450 6339.224
## 31       COMPASS_20220502_MSM_SipB_10cm 747.5236 5169.818
```

```r
# Constants needed for calculations:
clmw <- 35.45     #molecular weight of Chloride: 35.45
smw <- 32.06      #molecular weight of sulfur: 32.06


# Convert ppm to mmol/L
sampledat$SO4_mM <- (sampledat$SO4_ppm / smw)
sampledat$Cl_mM <- (sampledat$Cl_ppm / clmw)


# 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
sampledat$salinity <- ((1.8070 * sampledat$Cl_ppm) + 0.026) / 1000


head(sampledat)
```

```
##                               Sample_ID  SO4_ppm   Cl_ppm   SO4_mM    Cl_mM
## 26       COMPASS_20220502_MSM_SipA_10cm 924.7531 5530.741 28.84445 156.0153
## 27       COMPASS_20220502_MSM_SipA_20cm 686.1893 5431.449 21.40328 153.2144
## 28       COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549 12.46915 186.4753
## 29   COMPASS_20220502_MSM_SipA_45cm_Dup 398.1761 6525.085 12.41972 184.0645
```

```
## 30 COMPASS_20220502_MSM_SipA_45cm_Spike 606.2450 6339.224 18.90970 178.8215
## 31       COMPASS_20220502_MSM_SipB_10cm 747.5236 5169.818 23.31639 145.8341
##     salinity
## 26  9.994075
## 27  9.814655
## 28 11.945288
## 29 11.790855
## 30 11.455003
## 31  9.341887
```

Pull out dups and check with percent difference

```
#Show me the data that we have from the calculations
head(sampledat)
```

```
##                               Sample_ID  SO4_ppm   Cl_ppm   SO4_mM    Cl_mM
## 26       COMPASS_20220502_MSM_SipA_10cm 924.7531 5530.741 28.84445 156.0153
## 27       COMPASS_20220502_MSM_SipA_20cm 686.1893 5431.449 21.40328 153.2144
## 28       COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549 12.46915 186.4753
## 29   COMPASS_20220502_MSM_SipA_45cm_Dup 398.1761 6525.085 12.41972 184.0645
## 30 COMPASS_20220502_MSM_SipA_45cm_Spike 606.2450 6339.224 18.90970 178.8215
## 31       COMPASS_20220502_MSM_SipB_10cm 747.5236 5169.818 23.31639 145.8341
##     salinity
## 26  9.994075
## 27  9.814655
## 28 11.945288
## 29 11.790855
## 30 11.455003
## 31  9.341887
```

```
#pull out any rows that have "dup" in the SampleID column
dups <- sampledat %>%
  filter(str_detect(Sample_ID, "Dup"))      #have to change this to match data
head(dups)
```

```
##                            Sample_ID   SO4_ppm   Cl_ppm   SO4_mM    Cl_mM
## 1 COMPASS_20220502_MSM_SipA_45cm_Dup  398.1761 6525.085 12.41972 184.0645
## 2 COMPASS_20220505_GWI_SipA_20cm_Dup 1069.4710 8909.081 33.35842 251.3140
##   salinity
## 1 11.79085
## 2 16.09874
```

```
#remove these from sample dataframe in a new dataframe
sampledat2 <- sampledat %>%
  filter(!str_detect(Sample_ID, "Dup")) %>%
  filter(!str_detect(Sample_ID, "Spike"))
head(sampledat2)
```

```
##                       Sample_ID  SO4_ppm   Cl_ppm   SO4_mM    Cl_mM  salinity
## 1 COMPASS_20220502_MSM_SipA_10cm 924.7531 5530.741 28.84445 156.0153  9.994075
```

```
## 2 COMPASS_20220502_MSM_SipA_20cm 686.1893 5431.449 21.40328 153.2144  9.814655
## 3 COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549 12.46915 186.4753 11.945288
## 4 COMPASS_20220502_MSM_SipB_10cm 747.5236 5169.818 23.31639 145.8341  9.341887
## 5 COMPASS_20220502_MSM_SipB_20cm 706.2678 5130.657 22.02956 144.7294  9.271123
## 6 COMPASS_20220502_MSM_SipC_10cm 869.9081 5414.163 27.13375 152.7267  9.783419
```

```
#remove the dup from these IDs so we will have duplicate sample names
dups$Sample_ID<-gsub("_Dup","",as.character(dups$Sample_ID))
dups <- dups[ ,-c(2,3)]
colnames(dups) <- c('Sample_ID', 'SO4_mM_dup', "Cl_mM_dup", "salinity_dup")
head(dups)
```

```
##                       Sample_ID SO4_mM_dup Cl_mM_dup salinity_dup
## 1 COMPASS_20220502_MSM_SipA_45cm   12.41972  184.0645     11.79085
## 2 COMPASS_20220505_GWI_SipA_20cm   33.35842  251.3140     16.09874
```

```
#put it back together with the old data set and look for duplicates
QAdups <- merge(sampledat2, dups)
head(QAdups)
```

```
##                       Sample_ID   SO4_ppm   Cl_ppm   SO4_mM    Cl_mM salinity
## 1 COMPASS_20220502_MSM_SipA_45cm  399.7609 6610.549 12.46915 186.4753 11.94529
## 2 COMPASS_20220505_GWI_SipA_20cm 1068.0465 8895.024 33.31399 250.9175 16.07333
##   SO4_mM_dup Cl_mM_dup salinity_dup
## 1   12.41972  184.0645     11.79085
## 2   33.35842  251.3140     16.09874
```

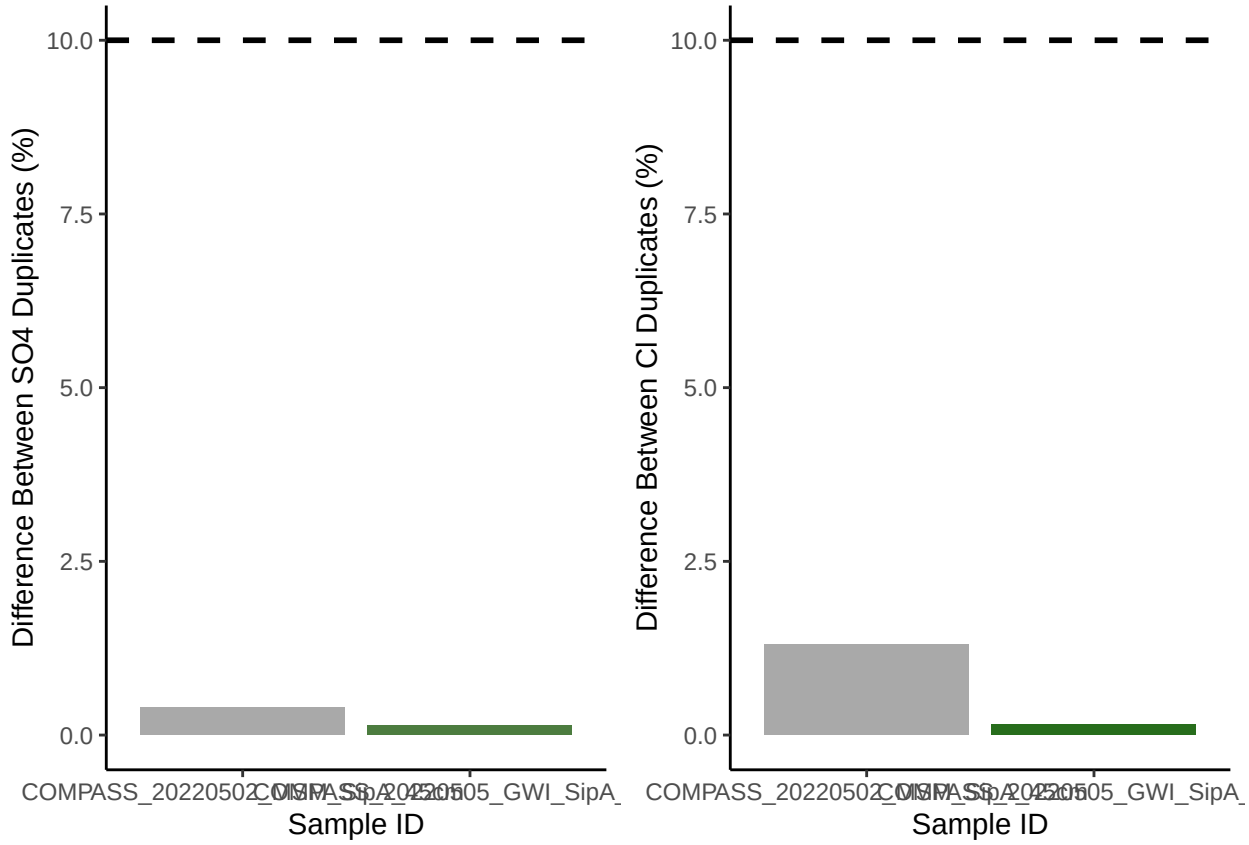
```
QAdups$SO4_dups_chk <- ((abs(QAdups$SO4_mM-QAdups$SO4_mM_dup))/((QAdups$SO4_mM+QAdups$SO4_mM_dup)/2))*1
QAdups$SO4_dups_flag <-  ifelse(QAdups$SO4_dups_chk <10, 'YES', 'NO, rerun')

QAdups$Cl_dups_chk <- ((abs(QAdups$Cl_mM-QAdups$Cl_mM_dup))/((QAdups$Cl_mM+QAdups$Cl_mM_dup)/2))*100
QAdups$Cl_dups_flag <-  ifelse(QAdups$Cl_dups_chk <10, 'YES', 'NO, rerun')

head(QAdups)
```

```
##                       Sample_ID   SO4_ppm   Cl_ppm   SO4_mM    Cl_mM salinity
## 1 COMPASS_20220502_MSM_SipA_45cm  399.7609 6610.549 12.46915 186.4753 11.94529
## 2 COMPASS_20220505_GWI_SipA_20cm 1068.0465 8895.024 33.31399 250.9175 16.07333
##   SO4_mM_dup Cl_mM_dup salinity_dup SO4_dups_chk SO4_dups_flag Cl_dups_chk
## 1   12.41972  184.0645     11.79085    0.3972243           YES   1.3012528
## 2   33.35842  251.3140     16.09874    0.1332855           YES   0.1579119
##   Cl_dups_flag
## 1          YES
## 2          YES
```

```
#plot dups output as a bar graph to easily check - want any over 10% to be red need to work on this
Sdupsbar <- ggplot(data = QAdups, aes(x = Sample_ID, y = SO4_dups_chk, fill=SO4_dups_chk)) +
       geom_bar(stat = 'identity') +
        scale_fill_gradient2(low='darkgreen', mid='darkgreen', high='darkgrey', space='Lab') +
        theme_classic() + labs(x= "Sample ID", y="Difference Between SO4 Duplicates (%)") +
        theme(legend.position="none") +  geom_hline(yintercept=10, linetype="dashed",
                color = "black", size=1)
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```
Cldupsbar <- ggplot(data = QAdups, aes(x = Sample_ID, y = Cl_dups_chk, fill=Cl_dups_chk)) +
       geom_bar(stat = 'identity') +
        scale_fill_gradient2(low='darkgreen', mid='darkgreen', high='darkgrey', space='Lab') +
        theme_classic() + labs(x= "Sample ID", y="Difference Between Cl Duplicates (%)") +
        theme(legend.position="none") +  geom_hline(yintercept=10, linetype="dashed",
                color = "black", size=1)

ggarrange(Sdupsbar, Cldupsbar,ncol=2, nrow=1)
```




```
#check for percent of no, reruns to see if it  would warrant reruns
Perc_dups <- QAdups %>%
  group_by(SO4_dups_flag) %>%
  summarise(S_no_rows = length(SO4_dups_flag))
Perc_dups1 <- QAdups %>%
  group_by(Cl_dups_flag) %>%
  summarise(Cl_no_rows = length(Cl_dups_flag))
colnames(Perc_dups) <- c("Flag", "S_no_rows")
colnames(Perc_dups1) <- c("Flag", "Cl_no_rows")
Perc_dups2 <- cbind(Perc_dups, Perc_dups1)
```

```
Perc_dups2$Total <- length(QAdups$SO4_dups_flag)
Perc_dups2$S_Percent <- (Perc_dups2$S_no_rows / Perc_dups2$Total)*100
Perc_dups2$Cl_Percent <- (Perc_dups2$Cl_no_rows / Perc_dups2$Total)*100
head(Perc_dups2)
```

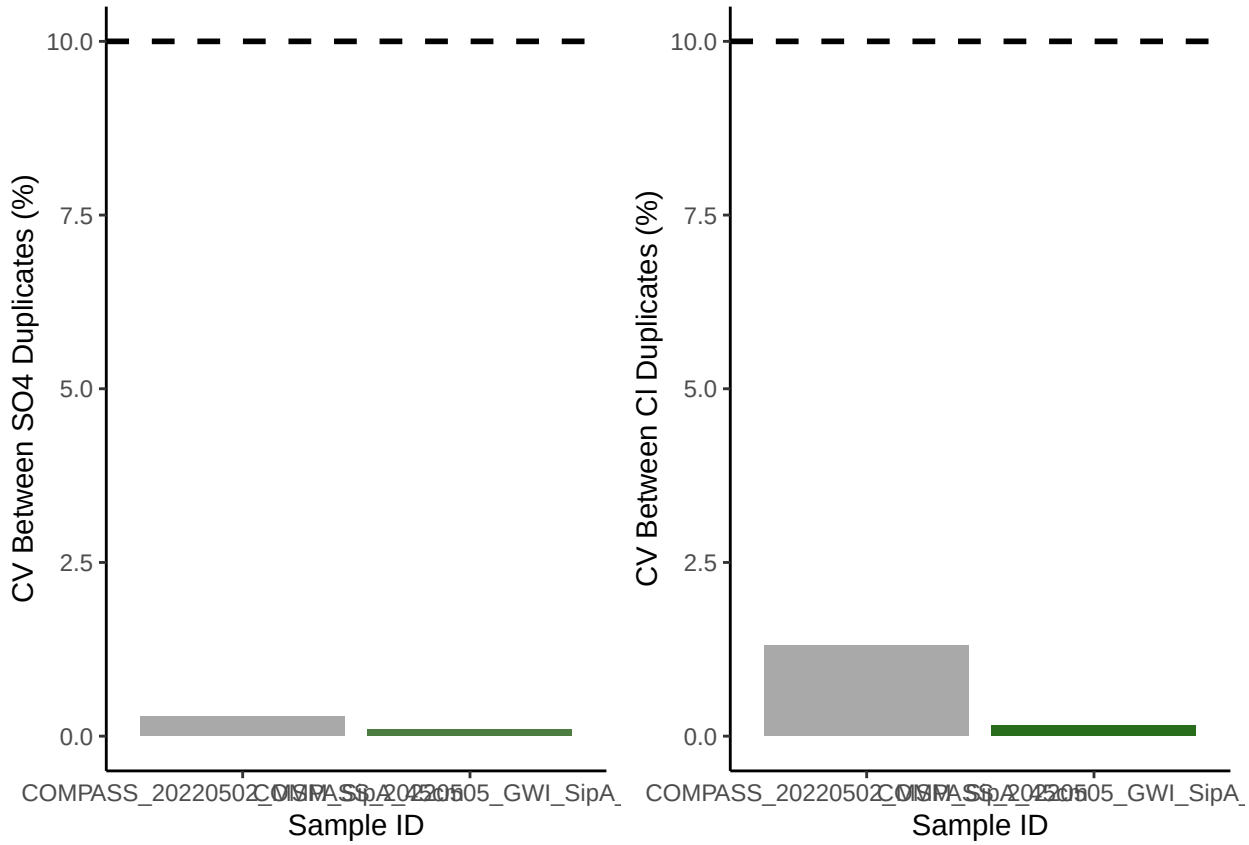
```
##   Flag S_no_rows Flag Cl_no_rows Total S_Percent Cl_Percent
## 1  YES         2  YES          2     2       100        100
```

Pull out dups and check with cv

```
#the cv
# calculate the sd bewteen the two duplicates then divide by the mean and multiply by 100

df2 <- as.data.frame(QAdups$SO4_mM)
df2$dups <- QAdups$SO4_mM_dup

df2$sds <- apply(df2,1,sd)

QAdups$SO4_dups_cv <- (df2$sds)/((QAdups$SO4_mM+QAdups$SO4_mM_dup)/2) * 100
QAdups$SO4_dups_cv_flag <-  ifelse(QAdups$SO4_dups_cv <11, 'YES', 'NO, rerun')

head(QAdups)
```

```
##                       Sample_ID   SO4_ppm   Cl_ppm   SO4_mM    Cl_mM salinity
## 1 COMPASS_20220502_MSM_SipA_45cm  399.7609 6610.549 12.46915 186.4753 11.94529
## 2 COMPASS_20220505_GWI_SipA_20cm 1068.0465 8895.024 33.31399 250.9175 16.07333
##   SO4_mM_dup Cl_mM_dup salinity_dup SO4_dups_chk SO4_dups_flag Cl_dups_chk
## 1   12.41972  184.0645     11.79085    0.3972243           YES   1.3012528
## 2   33.35842  251.3140     16.09874    0.1332855           YES   0.1579119
##   Cl_dups_flag SO4_dups_cv SO4_dups_cv_flag
## 1          YES  0.28088003              YES
## 2          YES  0.09424705              YES
```

```
#plot dups output as a bar graph to easily check - want any over 10% to be red need to work on this
Sdupsbar <- ggplot(data = QAdups, aes(x = Sample_ID, y = SO4_dups_cv, fill=SO4_dups_chk)) +
       geom_bar(stat = 'identity') +
        scale_fill_gradient2(low='darkgreen', mid='darkgreen', high='darkgrey', space='Lab') +
        theme_classic() + labs(x= "Sample ID", y="CV Between SO4 Duplicates (%)") +
        theme(legend.position="none") +  geom_hline(yintercept=10, linetype="dashed",
                color = "black", size=1)


Cldupsbar <- ggplot(data = QAdups, aes(x = Sample_ID, y = Cl_dups_chk, fill=Cl_dups_chk)) +
       geom_bar(stat = 'identity') +
        scale_fill_gradient2(low='darkgreen', mid='darkgreen', high='darkgrey', space='Lab') +
        theme_classic() + labs(x= "Sample ID", y="CV Between Cl Duplicates (%)") +
        theme(legend.position="none") +  geom_hline(yintercept=10, linetype="dashed",
                color = "black", size=1)

ggarrange(Sdupsbar, Cldupsbar,ncol=2, nrow=1)
```




```
#check for percent of no, reruns to see if it  would warrant reruns
Perc_dups <- QAdups %>%
  group_by(SO4_dups_flag) %>%
  summarise(S_no_rows = length(SO4_dups_flag))
Perc_dups1 <- QAdups %>%
  group_by(Cl_dups_flag) %>%
  summarise(Cl_no_rows = length(Cl_dups_flag))
colnames(Perc_dups) <- c("Flag", "S_no_rows")
colnames(Perc_dups1) <- c("Flag", "Cl_no_rows")
Perc_dups2 <- cbind(Perc_dups, Perc_dups1)

Perc_dups2$Total <- length(QAdups$SO4_dups_flag)
Perc_dups2$S_Percent <- (Perc_dups2$S_no_rows / Perc_dups2$Total)*100
Perc_dups2$Cl_Percent <- (Perc_dups2$Cl_no_rows / Perc_dups2$Total)*100
head(Perc_dups2)
```

```
##   Flag S_no_rows Flag Cl_no_rows Total S_Percent Cl_Percent
## 1  YES         2  YES          2     2       100        100
```

Pull out spikes and check

```
#Show me the data that we have from the calculations
head(sampledat)
```

```
##                                Sample_ID  SO4_ppm   Cl_ppm   SO4_mM    Cl_mM
## 26       COMPASS_20220502_MSM_SipA_10cm 924.7531 5530.741 28.84445 156.0153
## 27       COMPASS_20220502_MSM_SipA_20cm 686.1893 5431.449 21.40328 153.2144
## 28       COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549 12.46915 186.4753
## 29   COMPASS_20220502_MSM_SipA_45cm_Dup 398.1761 6525.085 12.41972 184.0645
## 30 COMPASS_20220502_MSM_SipA_45cm_Spike 606.2450 6339.224 18.90970 178.8215
## 31       COMPASS_20220502_MSM_SipB_10cm 747.5236 5169.818 23.31639 145.8341
##     salinity
## 26  9.994075
## 27  9.814655
## 28 11.945288
## 29 11.790855
## 30 11.455003
## 31  9.341887
```

```
#pull out any rows that have "spk" in the SampleID column
spks <- sampledat %>%
  filter(str_detect(Sample_ID, "Spike"))      #have to change this to match data
head(spks)
```

```
##                               Sample_ID SO4_ppm   Cl_ppm  SO4_mM    Cl_mM
## 1 COMPASS_20220502_MSM_SipA_45cm_Spike 606.245 6339.224 18.9097 178.8215
##   salinity
## 1   11.455
```

```
#remove the dup from these IDs so we will have duplicate sample names
spks$Sample_ID<-gsub("_Spike","",as.character(spks$Sample_ID))
spks <- spks[ ,-c(2,3, 5,6)]
colnames(spks) <- c('Sample_ID', 'SO4_mM_spk')
head(spks)
```

```
##                        Sample_ID SO4_mM_spk
## 1 COMPASS_20220502_MSM_SipA_45cm    18.9097
```

```
#put it back together with the old data set and look for duplicates
QAspks <- merge(sampledat, spks)
head(QAspks)
```

```
##                        Sample_ID  SO4_ppm   Cl_ppm   SO4_mM    Cl_mM salinity
## 1 COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549 12.46915 186.4753 11.94529
##   SO4_mM_spk
## 1    18.9097
```

```
#now we need to calculate the spike concentration and calculate the spike recovery
spkconc <- (250/smw)    # in mM
spkvol <- 10             # in uL
spkvol <- spkvol/1000000
#spike for these samples was 10uL of the 250mM standard
QAspks$SO4_spk_Conc <- (spkconc)*spkvol         # mmoles of SO4
head(QAspks)
```

```
##                         Sample_ID  SO4_ppm   Cl_ppm   SO4_mM    Cl_mM salinity
## 1 COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549 12.46915 186.4753 11.94529
##   SO4_mM_spk SO4_spk_Conc
## 1    18.9097 7.797879e-05
```

```
#need to determine dilution factors and initial amount of sample added
#if your samples are all the same dilution just use the first line
#for Steph / COMPASS this depends on the site so...
QAspks$Dilution <- 1
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_UP"), 50, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_TR"), 50, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_WC"), 100, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_UP"), 50, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_TR"), 50, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_WC"), 100, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_UP"), 100, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_TR"), 100, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_WC"), 200, QAspks$Dilution)
#QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_UP"), 100, QAspks$Dilution)
#QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_TR"), 100, QAspks$Dilution)
#QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_WC"), 100, QAspks$Dilution)

#Set Sample volumes in uL
QAspks$SampleVol <- 1
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_UP"), 1501, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_TR"), 1501, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_WC"), 1475, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_UP"), 1501, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_TR"), 1501, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_WC"), 1475, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_UP"), 1475, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_TR"), 1475, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_WC"), 1462, QAspks$SampleVol)
#QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_UP"), 100, QAspks$SampleVol)
#QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_TR"), 100, QAspks$SampleVol)
#QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_WC"), 100, QAspks$SampleVol)

#change sample volume to L
QAspks$SampleVol <- QAspks$SampleVol/1000000
head(QAspks)
```

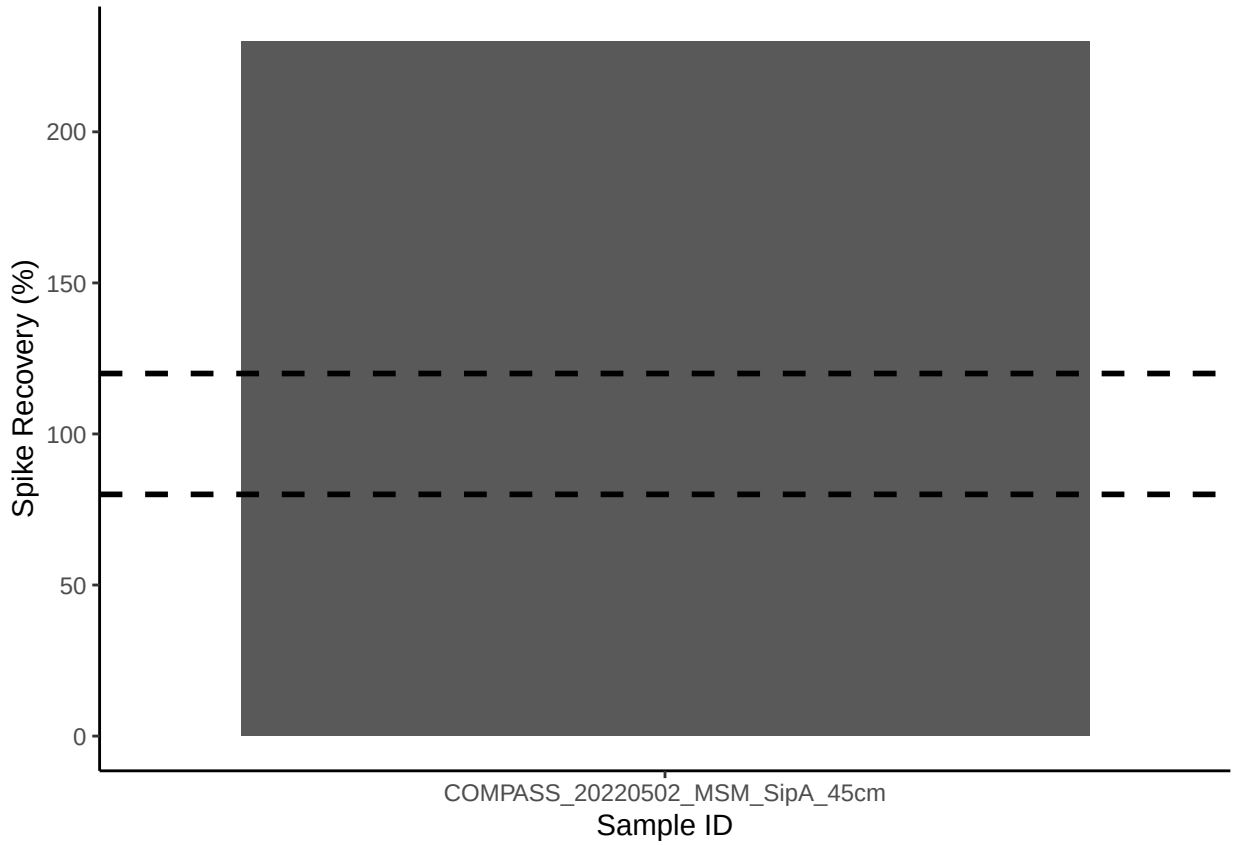
```
##                         Sample_ID  SO4_ppm   Cl_ppm   SO4_mM    Cl_mM salinity
## 1 COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549 12.46915 186.4753 11.94529
##   SO4_mM_spk SO4_spk_Conc Dilution SampleVol
## 1    18.9097 7.797879e-05        1     1e-06
```

```
#gives us the total SO4 in the sample in mmoles
QAspks$SO4_Total_unspkd <- (QAspks$SO4_mM/QAspks$Dilution)*(QAspks$SampleVol)

##total SO4 in spiked sample in mmoles
QAspks$SO4_Total_spkd <- (QAspks$SO4_mM_spk/QAspks$Dilution)*(QAspks$SampleVol+spkvol)

QAspks$SO4_expctd_spkd <-  (QAspks$SO4_Total_unspkd + QAspks$SO4_spk_Conc)
```

```
QAspks$spk_recovery <-    (QAspks$SO4_Total_spkd/QAspks$SO4_expctd_spkd)*100
QAspks$SO4_spks_flag <-  ifelse(QAspks$spk_recovery <=120 & QAspks$spk_recovery >=80 , 'YES', 'NO, rerun

head(QAspks)
```

```
##                        Sample_ID  SO4_ppm   Cl_ppm   SO4_mM    Cl_mM salinity
## 1 COMPASS_20220502_MSM_SipA_45cm 399.7609 6610.549 12.46915 186.4753 11.94529
##   SO4_mM_spk SO4_spk_Conc Dilution SampleVol SO4_Total_unspkd SO4_Total_spkd
## 1    18.9097 7.797879e-05        1     1e-06     1.246915e-05   0.0002080067
##   SO4_expctd_spkd spk_recovery SO4_spks_flag
## 1    9.044794e-05      229.974     NO, rerun
```

```
#plot spk recoveries output as a bar graph to easily check - want any over 10% to be red need to work o
spksbar <- ggplot(data = QAspks, aes(x = Sample_ID, y = spk_recovery)) +
        geom_bar(stat = 'identity') +
        scale_fill_manual(values=c("lightblue4")) +
        theme_classic() + labs(x= "Sample ID", y="Spike Recovery (%)") +
        theme(legend.position="none") +
        geom_hline(yintercept=80, linetype="dashed", color = "black", size=1) +
        geom_hline(yintercept=120, linetype="dashed", color = "black", size=1)

spksbar
```



```
#check for percent of no, reruns to see if it  would warrant reruns
Perc_spks <- QAspks %>%
  group_by(SO4_spks_flag) %>%
  summarise(no_rows = length(SO4_spks_flag))
Perc_spks$Total <- length(QAspks$SO4_spks_flag)
Perc_spks$Percent <- (Perc_spks$no_rows / Perc_spks$Total)*100
head(Perc_spks)
```

```
## # A tibble: 1 x 4
##   SO4_spks_flag no_rows Total Percent
##   <chr>           <int> <int>   <dbl>
## 1 NO, rerun           1     1     100
```

Make final dataframe with IDs

```
#for steph <- pull out identifiers of the sample names
#pull the sample ID and separate it by the underscores
IDs <- data.frame(do.call('rbind', strsplit(as.character(sampledat2$Sample_ID),'_',fixed=TRUE)))
colnames(IDs) <- c("Analysis_No" ,"Site", "Zone", "Replicate", "Depth")
head(IDs)
```

```
##   Analysis_No     Site Zone Replicate Depth
## 1     COMPASS 20220502  MSM      SipA  10cm
## 2     COMPASS 20220502  MSM      SipA  20cm
## 3     COMPASS 20220502  MSM      SipA  45cm
## 4     COMPASS 20220502  MSM      SipB  10cm
## 5     COMPASS 20220502  MSM      SipB  20cm
## 6     COMPASS 20220502  MSM      SipC  10cm
```

```
#rejoin them to the dataframe
alldat <- cbind(IDs, sampledat2)
head(alldat)
```

```
##   Analysis_No     Site Zone Replicate Depth                      Sample_ID
## 1     COMPASS 20220502  MSM      SipA  10cm COMPASS_20220502_MSM_SipA_10cm
## 2     COMPASS 20220502  MSM      SipA  20cm COMPASS_20220502_MSM_SipA_20cm
## 3     COMPASS 20220502  MSM      SipA  45cm COMPASS_20220502_MSM_SipA_45cm
## 4     COMPASS 20220502  MSM      SipB  10cm COMPASS_20220502_MSM_SipB_10cm
## 5     COMPASS 20220502  MSM      SipB  20cm COMPASS_20220502_MSM_SipB_20cm
## 6     COMPASS 20220502  MSM      SipC  10cm COMPASS_20220502_MSM_SipC_10cm
##    SO4_ppm   Cl_ppm   SO4_mM    Cl_mM  salinity
## 1 924.7531 5530.741 28.84445 156.0153  9.994075
## 2 686.1893 5431.449 21.40328 153.2144  9.814655
## 3 399.7609 6610.549 12.46915 186.4753 11.945288
## 4 747.5236 5169.818 23.31639 145.8341  9.341887
## 5 706.2678 5130.657 22.02956 144.7294  9.271123
## 6 869.9081 5414.163 27.13375 152.7267  9.783419
```

Make final dataframe with IDs

```
#setwd("S:/Biogeochemistry/People/Wilson (Steph)/Data/Dionex/Final Data Files")        #Change wd

write.csv(alldat, file="Processed Data/COMPASS_SynopticCB_PW_Processed_Cl_SO4_202205.csv")      #Change
```

END