Dionex_COMPASS_June2023

Stephanie J. Wilson

2023-06-23

Daily Set up

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

```
# SULFATE DATA:

## Read in raw data file from Dionex - copied and saved as a txt
Sdat <- read.table("Raw Data/COMPASS_Synoptic_CB_MonMon_202306_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 4.473   0.0159     3.37   0.0154     0.09
## 2 2  Lab Blank              Unknown 4.490   0.0155     3.53   0.0151     0.08
## 3 3  Lab Blank              Unknown 4.493   0.0156     3.53   0.0151     0.08
## 4 4  Lab Blank              Unknown 4.497   0.0163     3.62   0.0158     0.09
## 5 5 Standard 1 Calibration Standard 4.503   0.7135     9.66   0.6906     5.08
## 6 6 Standard 2 Calibration Standard 4.497   1.2518     8.87   1.2118     8.59
```

```
## 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  0.0159
## 2  Lab Blank  0.0155
## 3  Lab Blank  0.0156
## 4  Lab Blank  0.0163
## 5 Standard 1  0.7135
## 6 Standard 2  1.2518
```

```
## 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
```

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

```
##    Sample_ID SO4_ppm
## 1  Lab Blank  0.0159
## 2  Lab Blank  0.0155
## 3  Lab Blank  0.0156
## 4  Lab Blank  0.0163
## 5 Standard 1  0.7135
## 6 Standard 2  1.2518
```

```r
#Chloride data
## Read in raw data file from Dionex - copied and saved as a txt
Cldat <- read.table("Raw Data/COMPASS_Synoptic_CB_MonMon_202306_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 3.543  0.0226    6.28   0.0287    0.14
## 2 2  Lab Blank              Unknown 3.540  0.0096    2.86   0.0122    0.10
## 3 3  Lab Blank              Unknown 3.543  0.0095    2.81   0.0120    0.10
## 4 4  Lab Blank              Unknown 3.543  0.0099    2.88   0.0126    0.11
## 5 5 Standard 1 Calibration Standard 3.550  5.0880   90.34   6.4594   56.89
## 6 6 Standard 2 Calibration Standard 3.550  9.8032   91.13  12.4456  108.66
```

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

```
##          X.1 IC.Cl.1
## 1  Lab Blank 0.0226
## 2  Lab Blank 0.0096
## 3  Lab Blank 0.0095
## 4  Lab Blank 0.0099
## 5 Standard 1 5.0880
## 6 Standard 2 9.8032
```

```r
## 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 0.0226
## 2  Lab Blank 0.0096
## 3  Lab Blank 0.0095
## 4  Lab Blank 0.0099
## 5 Standard 1 5.0880
## 6 Standard 2 9.8032
## Bring the data back together:
all_dat <- merge(Sdat, Cldat, by="Sample_ID")
head(all_dat)
```

```
##   Sample_ID   SO4_ppm     Cl_ppm
## 1           1790.3015 13351.2461
## 2           1790.3015  4162.4856
## 3           1790.3015     0.0056
## 4           1790.3015  4096.1384
## 5           1790.3015         NA
## 6            462.8695 13351.2461
```

```
## 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       1_GCW_202306_UP_LysA_20cm    5.2274     7.1807
## 27      10_GCW_202306_TR_LysB_20cm    4.9649     6.2568
## 28  10_GCW_202306_TR_LysB_20cm_dup    5.0878     6.5878
## 29     100_GWI_202306_WC_LysA_10cm 1136.1780 11561.9880
## 30 100_GWI_202306_WC_LysA_10cm_dup 1192.1473 12081.0024
## 31     101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142
```

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

```
##                           Sample_ID   SO4_ppm     Cl_ppm
## 26       1_GCW_202306_UP_LysA_20cm    5.2274     7.1807
## 27      10_GCW_202306_TR_LysB_20cm    4.9649     6.2568
## 28  10_GCW_202306_TR_LysB_20cm_dup    5.0878     6.5878
## 29     100_GWI_202306_WC_LysA_10cm 1136.1780 11561.9880
## 30 100_GWI_202306_WC_LysA_10cm_dup 1192.1473 12081.0024
## 31     101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142
```

```
## If there is an n.a. in the dataframe that is because there was no peak, which
# would mean there was no Cl or SO4 and we want to know that so make all n.a.'s into zeros
all_dat[is.na(all_dat)] <- 0
```

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

```
stds <- all_dat[grepl("Standard", all_dat$Sample_ID),]
stds <- stds[-c(17),]
head(stds)
```

```
##      Sample_ID SO4_ppm Cl_ppm
## 621 Standard 1  0.7831 5.4746
## 622 Standard 1  0.7831 5.0880
## 623 Standard 1  0.7831 5.5265
## 624 Standard 1  0.7831 5.1245
## 625 Standard 1  0.7135 5.4746
## 626 Standard 1  0.7135 5.0880
```

```
stds_chk_S <- stds %>%
    group_by(Sample_ID) %>%
    summarise(mean = mean(SO4_ppm), sd = sd(SO4_ppm))
stds_chk_S$cv <- (stds_chk_S$sd/stds_chk_S$mean)*100
stds_chk_S$flag <-  ifelse(stds_chk_S$cv <5, 'YES', 'NO, rerun')
head(stds_chk_S)
```

```
## # A tibble: 5 x 5
##   Sample_ID    mean     sd    cv flag
##   <fct>       <dbl>  <dbl> <dbl> <chr>
## 1 Standard 1  0.757 0.0352  4.66 YES
## 2 Standard 2  1.33  0.0523  3.94 YES
## 3 Standard 3  2.33  0.0867  3.73 YES
## 4 Standard 4 10.7   0.293   2.74 YES
## 5 Standard 5 20.4   0.430   2.11 YES
```

```
stds_chk_Cl <- stds %>%
    group_by(Sample_ID) %>%
    summarise(mean = mean(Cl_ppm), sd = sd(Cl_ppm))
stds_chk_Cl$cv <- (stds_chk_Cl$sd/stds_chk_Cl$mean)*100
stds_chk_Cl$flag <-  ifelse(stds_chk_Cl$cv <5, 'YES', 'NO, rerun')
head(stds_chk_Cl)
```

```
## # A tibble: 5 x 5
##   Sample_ID    mean    sd    cv flag
##   <fct>       <dbl> <dbl> <dbl> <chr>
## 1 Standard 1   5.30 0.205  3.86 YES
## 2 Standard 2  10.3  0.382  3.71 YES
## 3 Standard 3  20.9  0.792  3.79 YES
## 4 Standard 4 104.   3.04   2.93 YES
## 5 Standard 5 206.   4.84   2.35 YES
```

Calculate mmol/L concentrations

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

```
##                           Sample_ID   SO4_ppm     Cl_ppm
## 26       1_GCW_202306_UP_LysA_20cm    5.2274     7.1807
## 27      10_GCW_202306_TR_LysB_20cm    4.9649     6.2568
## 28  10_GCW_202306_TR_LysB_20cm_dup    5.0878     6.5878
## 29     100_GWI_202306_WC_LysA_10cm 1136.1780 11561.9880
## 30 100_GWI_202306_WC_LysA_10cm_dup 1192.1473 12081.0024
## 31     101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142
```

```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       1_GWC_202306_UP_LysA_20cm    5.2274     7.1807  0.1630505   0.2025585
## 27      10_GWC_202306_TR_LysB_20cm    4.9649     6.2568  0.1548628   0.1764965
## 28  10_GWC_202306_TR_LysB_20cm_dup    5.0878     6.5878  0.1586962   0.1858336
## 29     100_GWI_202306_WC_LysA_10cm 1136.1780 11561.9880 35.4391142 326.1491678
## 30 100_GWI_202306_WC_LysA_10cm_dup 1192.1473 12081.0024 37.1848815 340.7899126
## 31     101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142 41.3467498 308.6097094
##       salinity
## 26  0.01300152
## 27  0.01133204
## 28  0.01193015
## 29 20.89253832
## 30 21.83039734
## 31 19.76899306
```

Pull out dups and check with percent difference

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

```
##                          Sample_ID   SO4_ppm     Cl_ppm     SO4_mM       Cl_mM
## 26       1_GWC_202306_UP_LysA_20cm    5.2274     7.1807  0.1630505   0.2025585
## 27      10_GWC_202306_TR_LysB_20cm    4.9649     6.2568  0.1548628   0.1764965
## 28  10_GWC_202306_TR_LysB_20cm_dup    5.0878     6.5878  0.1586962   0.1858336
## 29     100_GWI_202306_WC_LysA_10cm 1136.1780 11561.9880 35.4391142 326.1491678
## 30 100_GWI_202306_WC_LysA_10cm_dup 1192.1473 12081.0024 37.1848815 340.7899126
## 31     101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142 41.3467498 308.6097094
##       salinity
## 26  0.01300152
## 27  0.01133204
## 28  0.01193015
## 29 20.89253832
## 30 21.83039734
## 31 19.76899306
```

```
#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  10_GCW_202306_TR_LysB_20cm_dup    5.0878     6.5878  0.1586962   0.1858336
## 2 100_GWI_202306_WC_LysA_10cm_dup 1192.1473 12081.0024 37.1848815 340.7899126
## 3         110_GWI_202306_SW_C_dup 1716.1138 12692.8628 53.5281909 358.0497264
## 4  120_MSM_202306_WC_RHZ_Col1_dup 1093.4677  9805.4513 34.1069152 276.5994725
## 5     130_MSM_202036_UP_PPR_3_dup   77.8144   975.9884  2.4271491  27.5314076
## 6     140_MSM_202306_TR_PPR_1_dup  158.3756  1642.2791  4.9399750  46.3266319
##      salinity
## 1  0.01193015
## 2 21.83039734
## 3 22.93602908
## 4 17.71847650
## 5  1.76363704
## 6  2.96762433
```

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

```
##                     Sample_ID   SO4_ppm     Cl_ppm     SO4_mM       Cl_mM
## 1   1_GCW_202306_UP_LysA_20cm    5.2274     7.1807  0.1630505   0.2025585
## 2  10_GCW_202306_TR_LysB_20cm    4.9649     6.2568  0.1548628   0.1764965
## 3 100_GWI_202306_WC_LysA_10cm 1136.1780 11561.9880 35.4391142 326.1491678
## 4 101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142 41.3467498 308.6097094
## 5 102_GWI_202036_WC_LysA_45cm 1374.3504 11492.1490 42.8680724 324.1790973
## 6 103_GWI_202306_WC_LysB_10cm 1309.7705 12131.0664 40.8537274 342.2021551
##      salinity
## 1  0.01300152
## 2  0.01133204
## 3 20.89253832
## 4 19.76899306
## 5 20.76633924
## 6 21.92086298
```

```
#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  10_GCW_202306_TR_LysB_20cm  0.1586962   0.1858336   0.01193015
## 2 100_GWI_202306_WC_LysA_10cm 37.1848815 340.7899126  21.83039734
## 3         110_GWI_202306_SW_C 53.5281909 358.0497264  22.93602908
## 4  120_MSM_202306_WC_RHZ_Col1 34.1069152 276.5994725  17.71847650
## 5     130_MSM_202036_UP_PPR_3  2.4271491  27.5314076   1.76363704
## 6     140_MSM_202306_TR_PPR_1  4.9399750  46.3266319   2.96762433
```

```
#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
## 1  10_GCW_202306_TR_LysB_20cm    4.9649     6.2568  0.1548628   0.1764965
## 2 100_GWI_202306_WC_LysA_10cm 1136.1780 11561.9880 35.4391142 326.1491678
## 3  120_MSM_202306_WC_RHZ_Col1 1094.7124  9713.3353 34.1457392 274.0009958
## 4     140_MSM_202306_TR_PPR_1  164.8393  1705.2825  5.1415876  48.1038787
## 5    150_MSM_202306_TR_PPR_11  820.7486  6672.6120 25.6003930 188.2260085
## 6     160_MSM_202306_WC_PPR_9 1026.1569  8888.6427 32.0073893 250.7374528
##      salinity SO4_mM_dup   Cl_mM_dup salinity_dup
## 1  0.01133204  0.1586962   0.1858336   0.01193015
## 2 20.89253832 37.1848815 340.7899126  21.83039734
## 3 17.55202289 34.1069152 276.5994725  17.71847650
## 4  3.08147148  4.9399750  46.3266319   2.96762433
## 5 12.05743588 25.5830724 188.8710635  12.09875691
## 6 16.06180336 33.0801029 256.5228039  16.43240225
```

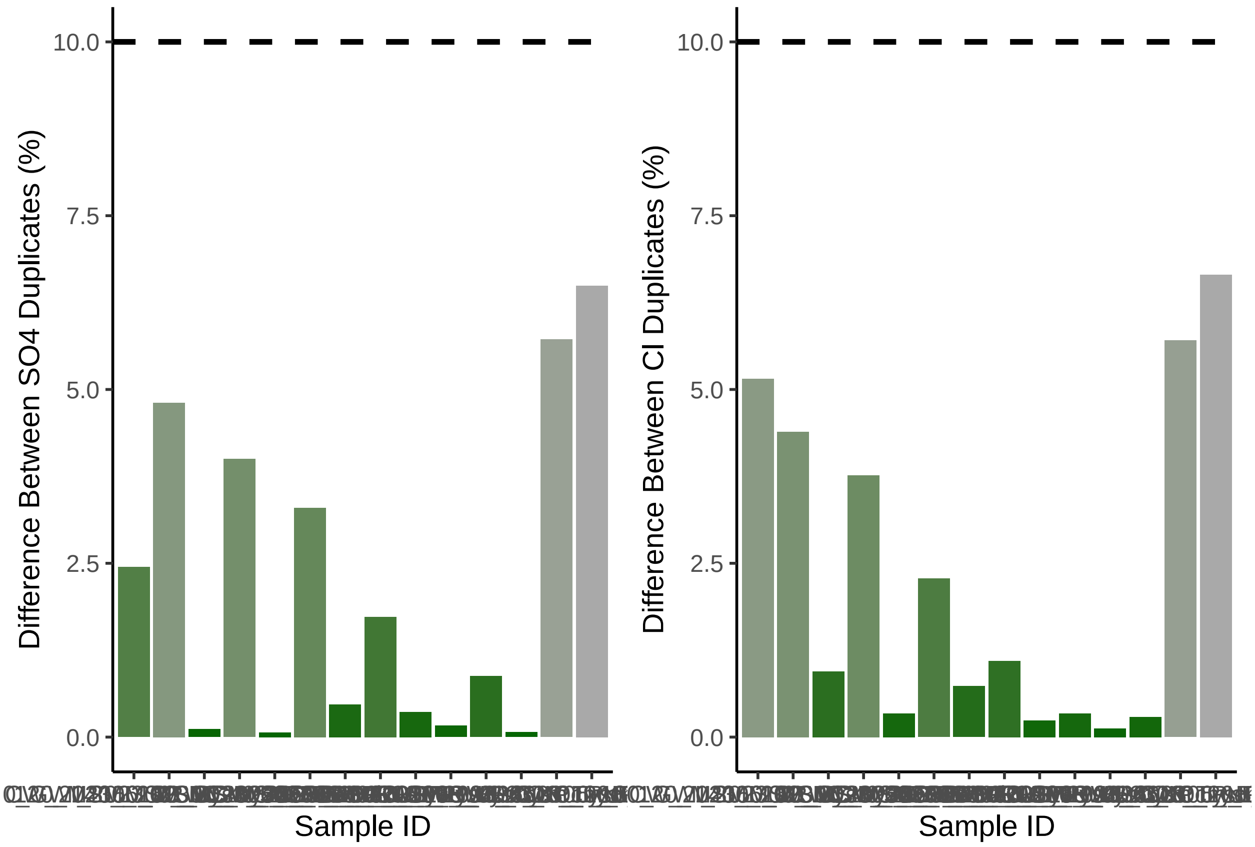
```
QAdups$SO4_dups_chk <- ((abs(QAdups$SO4_mM-QAdups$SO4_mM_dup))/((QAdups$SO4_mM+QAdups$SO4_mM_dup)/2))*1
QAdups$SO4_dups_flag <-  ifelse(QAdups$SO4_dups_chk <10, 'YES', 'NO, rerun')

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
## 1  10_GCW_202306_TR_LysB_20cm    4.9649     6.2568  0.1548628   0.1764965
## 2 100_GWI_202306_WC_LysA_10cm 1136.1780 11561.9880 35.4391142 326.1491678
## 3  120_MSM_202306_WC_RHZ_Col1 1094.7124  9713.3353 34.1457392 274.0009958
## 4     140_MSM_202306_TR_PPR_1  164.8393  1705.2825  5.1415876  48.1038787
## 5    150_MSM_202306_TR_PPR_11  820.7486  6672.6120 25.6003930 188.2260085
## 6     160_MSM_202306_WC_PPR_9 1026.1569  8888.6427 32.0073893 250.7374528
##      salinity SO4_mM_dup   Cl_mM_dup salinity_dup SO4_dups_chk SO4_dups_flag
## 1  0.01133204  0.1586962   0.1858336   0.01193015   2.44511425           YES
## 2 20.89253832 37.1848815 340.7899126  21.83039734   4.80768731           YES
## 3 17.55202289 34.1069152 276.5994725  17.71847650   0.11376577           YES
## 4  3.08147148  4.9399750  46.3266319   2.96762433   3.99962997           YES
## 5 12.05743588 25.5830724 188.8710635  12.09875691   0.06768064           YES
## 6 16.06180336 33.0801029 256.5228039  16.43240225   3.29622060           YES
##   Cl_dups_chk Cl_dups_flag
## 1   5.1539168          YES
## 2   4.3904294          YES
## 3   0.9438701          YES
## 4   3.7641369          YES
## 5   0.3421162          YES
## 6   2.2810189          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)
```

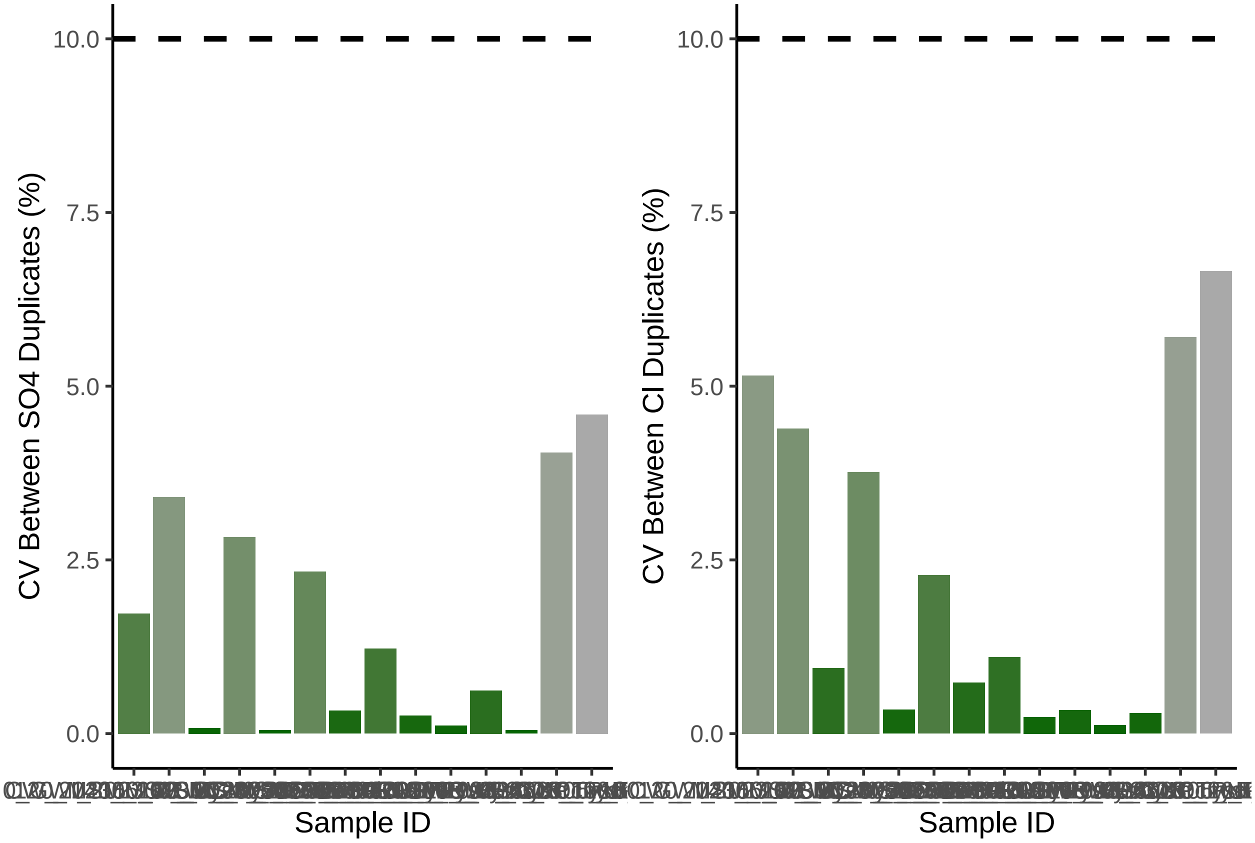
```
##   Flag S_no_rows Flag Cl_no_rows Total S_Percent Cl_Percent
## 1  YES        14  YES         14    14       100        100
```

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
## 1  10_GCW_202306_TR_LysB_20cm    4.9649     6.2568  0.1548628   0.1764965
## 2 100_GWI_202306_WC_LysA_10cm 1136.1780 11561.9880 35.4391142 326.1491678
## 3  120_MSM_202306_WC_RHZ_Col1 1094.7124  9713.3353 34.1457392 274.0009958
## 4     140_MSM_202306_TR_PPR_1  164.8393  1705.2825  5.1415876  48.1038787
## 5    150_MSM_202306_TR_PPR_11  820.7486  6672.6120 25.6003930 188.2260085
## 6     160_MSM_202306_WC_PPR_9 1026.1569  8888.6427 32.0073893 250.7374528
##      salinity SO4_mM_dup   Cl_mM_dup salinity_dup SO4_dups_chk SO4_dups_flag
## 1  0.01133204  0.1586962   0.1858336   0.01193015   2.44511425           YES
## 2 20.89253832 37.1848815 340.7899126  21.83039734   4.80768731           YES
## 3 17.55202289 34.1069152 276.5994725  17.71847650   0.11376577           YES
## 4  3.08147148  4.9399750  46.3266319   2.96762433   3.99962997           YES
## 5 12.05743588 25.5830724 188.8710635  12.09875691   0.06768064           YES
## 6 16.06180336 33.0801029 256.5228039  16.43240225   3.29622060           YES
##   Cl_dups_chk Cl_dups_flag SO4_dups_cv SO4_dups_cv_flag
## 1   5.1539168          YES  1.72895687              YES
## 2   4.3904294          YES  3.39954830              YES
## 3   0.9438701          YES  0.08044455             YES
## 4   3.7641369          YES  2.82816547             YES
## 5   0.3421162          YES  0.04785744             YES
## 6   2.2810189          YES  2.33077994             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        14  YES         14    14       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       1_GCW_202306_UP_LysA_20cm    5.2274     7.1807  0.1630505   0.2025585
## 27      10_GCW_202306_TR_LysB_20cm    4.9649     6.2568  0.1548628   0.1764965
## 28  10_GCW_202306_TR_LysB_20cm_dup    5.0878     6.5878  0.1586962   0.1858336
## 29     100_GWI_202306_WC_LysA_10cm 1136.1780 11561.9880 35.4391142 326.1491678
## 30 100_GWI_202306_WC_LysA_10cm_dup 1192.1473 12081.0024 37.1848815 340.7899126
## 31     101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142 41.3467498 308.6097094
##       salinity
## 26  0.01300152
## 27  0.01133204
## 28  0.01193015
## 29 20.89253832
## 30 21.83039734
## 31 19.76899306
```

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

```
##                         Sample_ID   SO4_ppm     Cl_ppm    SO4_mM       Cl_mM
## 1 101_GWI_202306_WC_LysA_20cm_spk 1614.9188 10400.4899 50.371765 293.3847645
## 2  11_GCW_202306_TR_LysB_45cm_spk   89.1234     4.2323  2.779894   0.1193879
## 3   111_MSM_202306_UP_RHZ_SF6_spk  222.0675  1072.2663  6.926622  30.2472863
## 4  121_MSM_202306_WC_RHZ_Col2_spk 1065.4168  9059.6735 33.231965 255.5620169
## 5     131_MSM_202306_UP_PPR_4_spk  186.3889   977.1367  5.813752  27.5637997
## 6     141_MSM_202306_TR_PPR_2_spk  324.6354  2338.5877 10.125870  65.9686234
##       salinity
## 1 18.793711249
## 2  0.007673766
## 3  1.937611204
## 4 16.370856014
## 5  1.765712017
## 6  4.225853974
```

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

```
##                      Sample_ID SO4_mM_spk
## 1 101_GWI_202306_WC_LysA_20cm  50.371765
## 2  11_GCW_202306_TR_LysB_45cm   2.779894
## 3   111_MSM_202306_UP_RHZ_SF6   6.926622
## 4  121_MSM_202306_WC_RHZ_Col2  33.231965
## 5     131_MSM_202306_UP_PPR_4   5.813752
## 6     141_MSM_202306_TR_PPR_2  10.125870
```

```
#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
## 1 101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142 41.3467498 308.609709
## 2  11_GCW_202306_TR_LysB_45cm    5.5728     4.2406  0.1738241   0.119622
## 3   111_MSM_202306_UP_RHZ_SF6  132.2880  1071.7023  4.1262633  30.231377
## 4  121_MSM_202306_WC_RHZ_Col2  875.2224  8937.4618 27.2995134 252.114578
## 5     131_MSM_202306_UP_PPR_4   99.0902  1030.9977  3.0907735  29.083151
## 6     141_MSM_202306_TR_PPR_2  237.8114  2340.9398  7.4176981  66.034973
##       salinity SO4_mM_spk
## 1 19.768993059  50.371765
## 2  0.007688764   2.779894
## 3  1.936592056   6.926622
## 4 16.150019473  33.231965
## 5  1.863038844   5.813752
## 6  4.230104219  10.125870
```

```
#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
## 1 101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142 41.3467498 308.609709
## 2  11_GCW_202306_TR_LysB_45cm    5.5728     4.2406  0.1738241   0.119622
## 3   111_MSM_202306_UP_RHZ_SF6  132.2880  1071.7023  4.1262633  30.231377
## 4  121_MSM_202306_WC_RHZ_Col2  875.2224  8937.4618 27.2995134 252.114578
## 5     131_MSM_202306_UP_PPR_4   99.0902  1030.9977  3.0907735  29.083151
## 6     141_MSM_202306_TR_PPR_2  237.8114  2340.9398  7.4176981  66.034973
##       salinity SO4_mM_spk SO4_spk_Conc
## 1 19.768993059  50.371765 7.797879e-05
## 2  0.007688764   2.779894 7.797879e-05
## 3  1.936592056   6.926622 7.797879e-05
## 4 16.150019473  33.231965 7.797879e-05
## 5  1.863038844   5.813752 7.797879e-05
## 6  4.230104219  10.125870 7.797879e-05
```

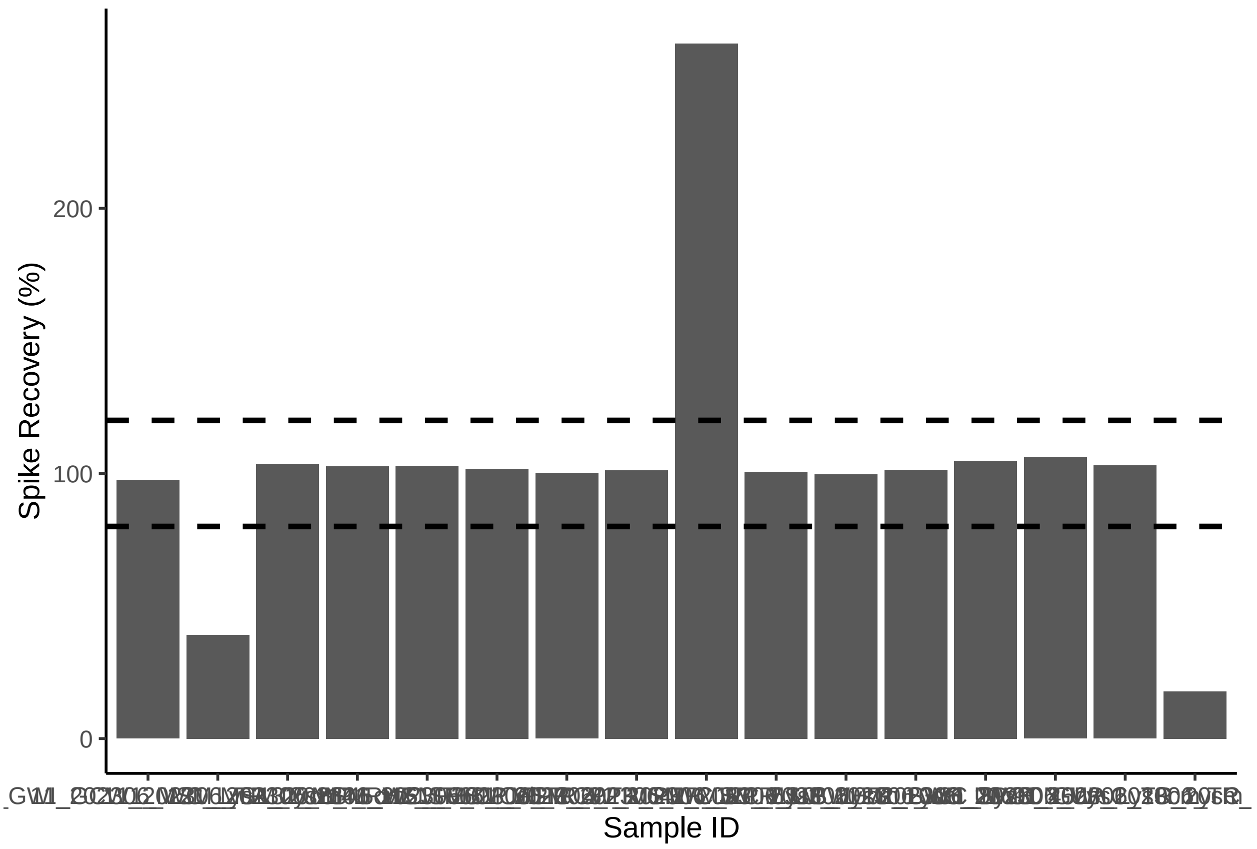
```r
#need to determine dilution factors and initial amount of sample added
#if your samples are all the same dilution just use the first line
#for Steph / COMPASS this depends on the site so...
QAspks$Dilution <- 1
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_202306_UP"), 50, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_202306_TR"), 50, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_202306_WC"), 100, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_202306_UP"), 50, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_202306_TR"), 50, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_202306_WC"), 100, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_202306_UP"), 100, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_202306_TR"), 100, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_202306_WC"), 200, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_202306_UP"), 50, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_202306_TR"), 50, QAspks$Dilution)
QAspks$Dilution <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_202306_WC"), 50, QAspks$Dilution)

#Set Sample volumes in uL
QAspks$SampleVol <- 1
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_202306_UP"), 1501, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_202306_TR"), 1501, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "MSM_202306_WC"), 1475, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_202306_UP"), 1501, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_202306_TR"), 1501, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GCrew_202306_WC"), 1475, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_202306_UP"), 1475, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_202306_TR"), 1475, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "GWI_202306_WC"), 1462, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_202306_UP"), 1501, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_202306_TR"), 1501, QAspks$SampleVol)
QAspks$SampleVol <-  ifelse(str_detect(QAspks$Sample_ID, "SWH_202306_WC"), 1501, QAspks$SampleVol)

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

```
##                     Sample_ID   SO4_ppm     Cl_ppm     SO4_mM      Cl_mM
## 1 101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142 41.3467498 308.609709
## 2  11_GCW_202306_TR_LysB_45cm    5.5728     4.2406  0.1738241   0.119622
## 3   111_MSM_202306_UP_RHZ_SF6  132.2880  1071.7023  4.1262633  30.231377
## 4  121_MSM_202306_WC_RHZ_Col2  875.2224  8937.4618 27.2995134 252.114578
## 5     131_MSM_202306_UP_PPR_4   99.0902  1030.9977  3.0907735  29.083151
## 6     141_MSM_202306_TR_PPR_2  237.8114  2340.9398  7.4176981  66.034973
##       salinity SO4_mM_spk SO4_spk_Conc Dilution SampleVol
## 1 19.768993059  50.371765 7.797879e-05      200  0.001462
## 2  0.007688764   2.779894 7.797879e-05        1  0.000001
## 3  1.936592056   6.926622 7.797879e-05       50  0.001501
## 4 16.150019473  33.231965 7.797879e-05      100  0.001475
## 5  1.863038844   5.813752 7.797879e-05       50  0.001501
## 6  4.230104219  10.125870 7.797879e-05       50  0.001501
```

```
#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, reru

head(QAspks)
```

```
##                     Sample_ID   SO4_ppm     Cl_ppm     SO4_mM      Cl_mM
## 1 101_GWI_202306_WC_LysA_20cm 1325.5768 10940.2142 41.3467498 308.609709
## 2  11_GCW_202306_TR_LysB_45cm    5.5728     4.2406  0.1738241   0.119622
## 3   111_MSM_202306_UP_RHZ_SF6  132.2880  1071.7023  4.1262633  30.231377
## 4  121_MSM_202306_WC_RHZ_Col2  875.2224  8937.4618 27.2995134 252.114578
## 5      131_MSM_202306_UP_PPR_4   99.0902  1030.9977  3.0907735  29.083151
## 6      141_MSM_202306_TR_PPR_2  237.8114  2340.9398  7.4176981  66.034973
##       salinity SO4_mM_spk SO4_spk_Conc Dilution SampleVol SO4_Total_unspkd
## 1 19.768993059  50.371765 7.797879e-05      200  0.001462     3.022447e-04
## 2  0.007688764   2.779894 7.797879e-05        1  0.000001     1.738241e-07
## 3  1.936592056   6.926622 7.797879e-05       50  0.001501     1.238704e-04
## 4 16.150019473  33.231965 7.797879e-05      100  0.001475     4.026678e-04
## 5  1.863038844   5.813752 7.797879e-05       50  0.001501     9.278502e-05
## 6  4.230104219  10.125870 7.797879e-05       50  0.001501     2.226793e-04
##   SO4_Total_spkd SO4_expctd_spkd spk_recovery SO4_spks_flag
## 1   3.707362e-04    3.802235e-04     97.50480           YES
## 2   3.057883e-05    7.815261e-05     39.12708     NO, rerun
## 3   2.093225e-04    2.018492e-04    103.70242           YES
## 4   4.934947e-04    4.806466e-04    102.67308           YES
## 5   1.756916e-04    1.707638e-04    102.88573           YES
## 6   3.060038e-04    3.006581e-04    101.77800           YES
```

```
#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: 2 x 4
##   SO4_spks_flag no_rows Total Percent
##   <chr>           <int> <int>   <dbl>
## 1 NO, rerun           3    16    18.8
## 2 YES                13    16    81.2
```

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)))
```

```
## Warning in rbind(c("1", "GCW", "202306", "UP", "LysA", "20cm"), c("10", :
## number of columns of result is not a multiple of vector length (arg 11)
```

```r
colnames(IDs) <- c("Analysis_No" ,"Site", "Date", "Zone", "Replicate", "Depth")
head(IDs)
```

```
##   Analysis_No Site   Date Zone Replicate Depth
## 1           1  GCW 202306   UP      LysA  20cm
## 2          10  GCW 202306   TR      LysB  20cm
## 3         100  GWI 202306   WC      LysA  10cm
## 4         101  GWI 202306   WC      LysA  20cm
## 5         102  GWI 202036   WC      LysA  45cm
## 6         103  GWI 202306   WC      LysB  10cm
```

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

```
##   Analysis_No Site   Date Zone Replicate Depth                   Sample_ID
## 1           1  GCW 202306   UP      LysA  20cm   1_GCW_202306_UP_LysA_20cm
## 2          10  GCW 202306   TR      LysB  20cm  10_GCW_202306_TR_LysB_20cm
## 3         100  GWI 202306   WC      LysA  10cm 100_GWI_202306_WC_LysA_10cm
## 4         101  GWI 202306   WC      LysA  20cm 101_GWI_202306_WC_LysA_20cm
## 5         102  GWI 202036   WC      LysA  45cm 102_GWI_202036_WC_LysA_45cm
## 6         103  GWI 202306   WC      LysB  10cm 103_GWI_202306_WC_LysB_10cm
##     SO4_ppm     Cl_ppm     SO4_mM       Cl_mM    salinity
## 1    5.2274     7.1807  0.1630505   0.2025585  0.01300152
## 2    4.9649     6.2568  0.1548628   0.1764965  0.01133204
## 3 1136.1780 11561.9880 35.4391142 326.1491678 20.89253832
## 4 1325.5768 10940.2142 41.3467498 308.6097094 19.76899306
## 5 1374.3504 11492.1490 42.8680724 324.1790973 20.76633924
## 6 1309.7705 12131.0664 40.8537274 342.2021551 21.92086298
```

Make final dataframe with IDs

```r
write.csv(alldat, file="Processed Data/COMPASS_SynopticCB_PW_Processed_Cl_SO4_202306.csv")             #C
```

END