Dionex_COMPASS_June2023

Stephanie J. Wilson

2023-06-23

Daily Set up

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

```r
# SULFATE DATA:

## Read in raw data file from Dionex - copied and saved as a txt
Sdat <- read.table("Raw Data/COMPASS_Synoptic_CB_MonMon_202308_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 4.927   0.4941     6.78   0.5941     3.77
## 6 6 Standard 2 Calibration Standard 4.913   1.0082     6.92   1.2123     7.67
```

```r
## 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   0.4941
## 6 Standard 2   1.0082
```

```r
## 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  0.4941
## 6 Standard 2  1.0082
```

```
#Chloride data
## Read in raw data file from Dionex - copied and saved as a txt
Cldat <- read.table("Raw Data/COMPASS_Synoptic_CB_MonMon_202308_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 3.870   0.0065   24.90   0.0104     0.05
## 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 3.900   5.0808   93.10   8.1547    64.88
## 6 6 Standard 2 Calibration Standard 3.893  10.1597   93.08  16.3065   129.27
```

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

```
##          X.1  IC.Cl.1
## 1  Lab Blank  0.0065
## 2  Lab Blank     n.a.
## 3  Lab Blank     n.a.
## 4  Lab Blank     n.a.
## 5 Standard 1  5.0808
## 6 Standard 2 10.1597
```

```
## 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
```

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

```
##     Sample_ID  Cl_ppm
## 1  Lab Blank  0.0065
## 2  Lab Blank      NA
## 3  Lab Blank      NA
## 4  Lab Blank      NA
## 5 Standard 1  5.0808
## 6 Standard 2 10.1597
```

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

```
##   Sample_ID   SO4_ppm     Cl_ppm
## 1             2718.7356 13626.7307
## 2             2718.7356  3949.4534
## 3             2718.7356     0.0065
## 4             2718.7356  4142.2056
## 5             2718.7356        NA
## 6              463.3476 13626.7307
```

```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        1_MSM_202308_UP_LysA_20cm  149.8293 1634.498
## 27       10_MSM_202308_TR_LysA_45cm  507.2198 4042.257
## 28  10_MSM_202308_TR_LysA_45cm_dup  489.5620 3887.412
## 29      100_GWI_202308_TR_LysB_45cm 1004.2827 8516.909
## 30  100_GWI_202308_TR_LysB_45cm_dup  996.3197 8400.961
## 31      101_GWI_202308_TR_LysC_10cm 1119.1039 7253.308
```

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

## 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
head(all_dat)
```

```
##                          Sample_ID   SO4_ppm   Cl_ppm
## 26        1_MSM_202308_UP_LysA_20cm  149.8293 1634.498
## 27       10_MSM_202308_TR_LysA_45cm  507.2198 4042.257
## 28  10_MSM_202308_TR_LysA_45cm_dup  489.5620 3887.412
## 29      100_GWI_202308_TR_LysB_45cm 1004.2827 8516.909
## 30  100_GWI_202308_TR_LysB_45cm_dup  996.3197 8400.961
## 31      101_GWI_202308_TR_LysC_10cm 1119.1039 7253.308
```

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

```r
stds <- all_dat[grepl("Standard", all_dat$Sample_ID),]
#stds <- stds[-c(17),]   #this is if you need to remove one for any reason
head(stds)
```

```
##        Sample_ID SO4_ppm Cl_ppm
## 582 Standard 1   0.4926 5.2343
## 583 Standard 1   0.4926 5.1521
## 584 Standard 1   0.4926 5.3688
```

```
## 585 Standard 1  0.4926 5.3987
## 586 Standard 1  0.4926 5.0808
## 587 Standard 1  0.4984 5.2343
```

```r
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  0.501 0.00778  1.55 YES
## 2 Standard 2  1.04  0.0264   2.55 YES
## 3 Standard 3  2.05  0.0514   2.51 YES
## 4 Standard 4 10.6   0.276    2.61 YES
## 5 Standard 5 20.7   0.536    2.59 YES
```

```r
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   5.25 0.125  2.38 YES
## 2 Standard 2  10.5  0.263  2.50 YES
## 3 Standard 3  20.5  0.580  2.83 YES
## 4 Standard 4 105.   2.89   2.75 YES
## 5 Standard 5 208.   5.74   2.76 YES
```

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      1_MSM_202308_UP_LysA_20cm  149.8293 1634.498
## 27     10_MSM_202308_TR_LysA_45cm  507.2198 4042.257
## 28 10_MSM_202308_TR_LysA_45cm_dup  489.5620 3887.412
## 29    100_GWI_202308_TR_LysB_45cm 1004.2827 8516.909
## 30 100_GWI_202308_TR_LysB_45cm_dup  996.3197 8400.961
## 31    101_GWI_202308_TR_LysC_10cm 1119.1039 7253.308
```

```
# 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_MSM_202308_UP_LysA_20cm  149.8293 1634.498   4.673403   46.10715
## 27        10_MSM_202308_TR_LysA_45cm  507.2198 4042.257  15.820954  114.02699
## 28  10_MSM_202308_TR_LysA_45cm_dup   489.5620 3887.412  15.270181  109.65903
## 29      100_GWI_202308_TR_LysB_45cm 1004.2827 8516.909  31.325100  240.25132
## 30 100_GWI_202308_TR_LysB_45cm_dup   996.3197 8400.961  31.076722  236.98057
## 31      101_GWI_202308_TR_LysC_10cm 1119.1039 7253.308  34.906547  204.60671
##      salinity
## 26   2.953565
## 27   7.304384
## 28   7.024580
## 29  15.390081
## 30  15.180563
## 31  13.106753
```

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         1_MSM_202308_UP_LysA_20cm  149.8293 1634.498   4.673403   46.10715
## 27        10_MSM_202308_TR_LysA_45cm  507.2198 4042.257  15.820954  114.02699
## 28  10_MSM_202308_TR_LysA_45cm_dup   489.5620 3887.412  15.270181  109.65903
## 29      100_GWI_202308_TR_LysB_45cm 1004.2827 8516.909  31.325100  240.25132
## 30 100_GWI_202308_TR_LysB_45cm_dup   996.3197 8400.961  31.076722  236.98057
## 31      101_GWI_202308_TR_LysC_10cm 1119.1039 7253.308  34.906547  204.60671
##      salinity
## 26   2.953565
## 27   7.304384
## 28   7.024580
```

```
## 29 15.390081
## 30 15.180563
## 31 13.106753
```

```r
#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_MSM_202308_TR_LysA_45cm_dup  489.5620  3887.412 15.27018 109.6590
## 2      100_GWI_202308_TR_LysB_45cm_dup  996.3197  8400.961 31.07672 236.9806
## 3      110_GWI_202308_WC_LysC_10cm_dup 1397.3222 12585.859 43.58460 355.0313
## 4 120_MSM_202308_TR_RHZ_SF_Tree_6_dup  703.8496  7147.621 21.95414 201.6254
## 5         129_MSM_202308_WC_RHZ_LysA_dup 1137.4814  9556.200 35.47977 269.5684
## 6        20_MSM_202308_WC_LysB_10cm_dup  804.7738  9753.018 25.10211 275.1204
##   salinity
## 1  7.02458
## 2 15.18056
## 3 22.74267
## 4 12.91578
## 5 17.26808
## 6 17.62373
```

```r
#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  salinity
## 1   1_MSM_202308_UP_LysA_20cm  149.8293 1634.498  4.673403  46.10715  2.953565
## 2  10_MSM_202308_TR_LysA_45cm  507.2198 4042.257 15.820954 114.02699  7.304384
## 3 100_GWI_202308_TR_LysB_45cm 1004.2827 8516.909 31.325100 240.25132 15.390081
## 4 101_GWI_202308_TR_LysC_10cm 1119.1039 7253.308 34.906547 204.60671 13.106753
## 5 102_GWI_202308_TR_LysC_20cm  892.7210 8024.734 27.845321 226.36766 14.500720
## 6 103_GWI_202308_TR_LysC_45cm 1093.3388 8828.044 34.102895 249.02805 15.952302
```

```r
#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_MSM_202308_TR_LysA_45cm   15.27018  109.6590      7.02458
## 2      100_GWI_202308_TR_LysB_45cm   31.07672  236.9806     15.18056
## 3      110_GWI_202308_WC_LysC_10cm   43.58460  355.0313     22.74267
## 4 120_MSM_202308_TR_RHZ_SF_Tree_6   21.95414  201.6254     12.91578
## 5       129_MSM_202308_WC_RHZ_LysA   35.47977  269.5684     17.26808
## 6       20_MSM_202308_WC_LysB_10cm   25.10211  275.1204     17.62373
```

```r
#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_MSM_202308_TR_LysA_45cm  507.2198  4042.257 15.82095 114.0270
## 2      100_GWI_202308_TR_LysB_45cm 1004.2827  8516.909 31.32510 240.2513
## 3      110_GWI_202308_WC_LysC_10cm 1402.0502 12627.728 43.73207 356.2123
## 4 120_MSM_202308_TR_RHZ_SF_Tree_6  701.8313  7146.360 21.89118 201.5899
## 5       129_MSM_202308_WC_RHZ_LysA 1157.9589  9769.314 36.11849 275.5801
## 6       20_MSM_202308_WC_LysB_10cm  801.5951  9723.497 25.00297 274.2877
##    salinity SO4_mM_dup Cl_mM_dup salinity_dup
## 1  7.304384   15.27018  109.6590      7.02458
## 2 15.390081   31.07672  236.9806     15.18056
## 3 22.818330   43.58460  355.0313     22.74267
## 4 12.913499   21.95414  201.6254     12.91578
## 5 17.653176   35.47977  269.5684     17.26808
## 6 17.570385   25.10211  275.1204     17.62373
```

```r
QAdups$SO4_dups_chk <- ((abs(QAdups$SO4_mM-QAdups$SO4_mM_dup))/((QAdups$SO4_mM+QAdups$SO4_mM_dup)/2))*100
QAdups$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_MSM_202308_TR_LysA_45cm  507.2198  4042.257 15.82095 114.0270
## 2      100_GWI_202308_TR_LysB_45cm 1004.2827  8516.909 31.32510 240.2513
## 3      110_GWI_202308_WC_LysC_10cm 1402.0502 12627.728 43.73207 356.2123
## 4 120_MSM_202308_TR_RHZ_SF_Tree_6  701.8313  7146.360 21.89118 201.5899
## 5       129_MSM_202308_WC_RHZ_LysA 1157.9589  9769.314 36.11849 275.5801
## 6       20_MSM_202308_WC_LysB_10cm  801.5951  9723.497 25.00297 274.2877
##    salinity SO4_mM_dup Cl_mM_dup salinity_dup SO4_dups_chk SO4_dups_flag
## 1  7.304384   15.27018  109.6590      7.02458    3.5429620           YES
## 2 15.390081   31.07672  236.9806     15.18056    0.7960602           YES
## 3 22.818330   43.58460  355.0313     22.74267    0.3377900           YES
## 4 12.913499   21.95414  201.6254     12.91578    0.2871633           YES
## 5 17.653176   35.47977  269.5684     17.26808    1.7841893           YES
## 6 17.570385   25.10211  275.1204     17.62373    0.3957621           YES
##   Cl_dups_chk Cl_dups_flag
## 1   3.9054416          YES
## 2   1.3707174          YES
## 3   0.3321114          YES
## 4   0.0176312          YES
## 5   2.2055207          YES
## 6   0.3031415          YES
```

```r
#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') +
```
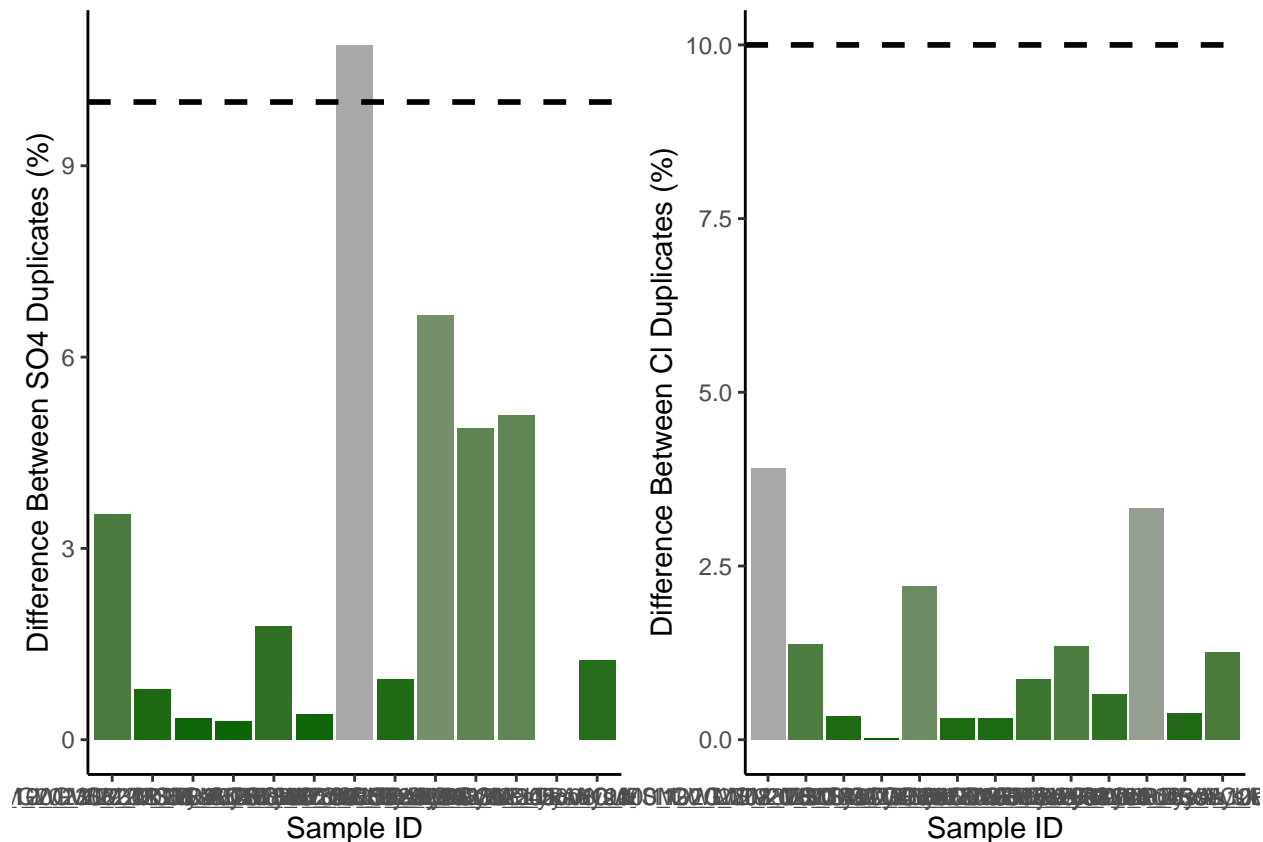
```
        scale_fill_gradient2(low='darkgreen', mid='darkgreen', high='darkgrey', space='Lab') +
        theme_classic() + labs(x= "Sample ID", y="Difference Between SO4 Duplicates (%)") +
        theme(legend.position="none") + geom_hline(yintercept=10, linetype="dashed",
              color = "black", 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)
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## ('geom_bar()').
```

```r
#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 NO, rerun         1  YES         13    13  7.692308        100
## 2       YES        11  YES         13    13 84.615385        100
## 3      <NA>         1  YES         13    13  7.692308        100
```

Pull out dups and check with cv

```r
#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_MSM_202308_TR_LysA_45cm  507.2198  4042.257 15.82095 114.0270
## 2      100_GWI_202308_TR_LysB_45cm 1004.2827  8516.909 31.32510 240.2513
## 3      110_GWI_202308_WC_LysC_10cm 1402.0502 12627.728 43.73207 356.2123
## 4 120_MSM_202308_TR_RHZ_SF_Tree_6  701.8313  7146.360 21.89118 201.5899
## 5       129_MSM_202308_WC_RHZ_LysA 1157.9589  9769.314 36.11849 275.5801
## 6       20_MSM_202308_WC_LysB_10cm  801.5951  9723.497 25.00297 274.2877
##    salinity SO4_mM_dup Cl_mM_dup salinity_dup SO4_dups_chk SO4_dups_flag
## 1  7.304384   15.27018  109.6590      7.02458    3.5429620           YES
## 2 15.390081   31.07672  236.9806     15.18056    0.7960602           YES
## 3 22.818330   43.58460  355.0313     22.74267    0.3377900           YES
## 4 12.913499   21.95414  201.6254     12.91578    0.2871633           YES
## 5 17.653176   35.47977  269.5684     17.26808    1.7841893           YES
## 6 17.570385   25.10211  275.1204     17.62373    0.3957621           YES
##   Cl_dups_chk Cl_dups_flag SO4_dups_cv SO4_dups_cv_flag
```

```
## 1    3.9054416          YES    2.5052524            YES
## 2    1.3707174          YES    0.5628996            YES
## 3    0.3321114          YES    0.2388536            YES
## 4    0.0176312          YES    0.2030551            YES
## 5    2.2055207          YES    1.2616123            YES
## 6    0.3031415          YES    0.2798461            YES
```
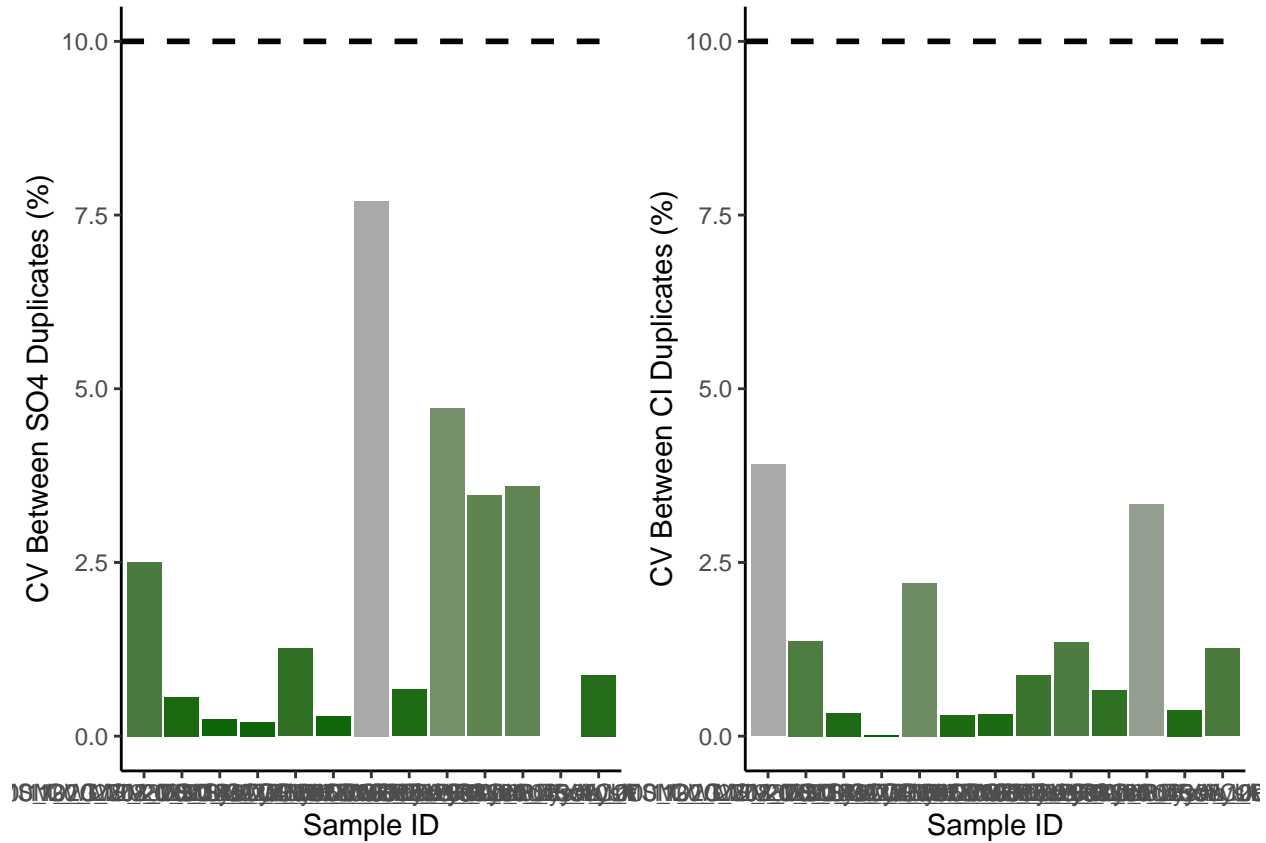
```r
#plot dups output as a bar graph to easily check - want any over 10% to be red need to work on this
Sdupsbar <- ggplot(data = QAdups, aes(x = Sample_ID, y = SO4_dups_cv, fill=SO4_dups_chk)) +
      geom_bar(stat = 'identity') +
       scale_fill_gradient2(low='darkgreen', mid='darkgreen', high='darkgrey', 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)
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## ('geom_bar()').
```

```
#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 NO, rerun         1  YES         13    13  7.692308        100
## 2       YES        11  YES         13    13 84.615385        100
## 3      <NA>         1  YES         13    13  7.692308        100
```

Pull out spikes and check

```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_MSM_202308_UP_LysA_20cm  149.8293 1634.498  4.673403  46.10715
## 27       10_MSM_202308_TR_LysA_45cm  507.2198 4042.257 15.820954 114.02699
## 28   10_MSM_202308_TR_LysA_45cm_dup  489.5620 3887.412 15.270181 109.65903
## 29       100_GWI_202308_TR_LysB_45cm 1004.2827 8516.909 31.325100 240.25132
## 30 100_GWI_202308_TR_LysB_45cm_dup  996.3197 8400.961 31.076722 236.98057
## 31       101_GWI_202308_TR_LysC_10cm 1119.1039 7253.308 34.906547 204.60671
##    salinity
## 26  2.953565
## 27  7.304384
## 28  7.024580
## 29 15.390081
## 30 15.180563
## 31 13.106753
```

```r
#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_202308_TR_LysC_10cm_spk 1303.8902  7224.032 40.67031 203.7809
## 2       11_MSM_202308_TR_LysB_10cm_spk  601.3867  5567.455 18.75816 157.0509
## 3      111_GWI_202308_WC_LysC_20cm_spk 1387.3085 11728.317 43.27226 330.8411
## 4 121_MSM_202308_TR_RHZ_SF_Tree_7_spk  793.1324  7277.755 24.73900 205.2963
## 5       130_MSM_202308_WC_RHZ_LysC_spk  994.0336  9577.230 31.00541 270.1616
## 6       21_MSM_202308_WC_LysB_20cm_spk  883.5257  8773.534 27.55851 247.4904
##    salinity
## 1 13.05385
## 2 10.06042
## 3 21.19309
## 4 13.15093
## 5 17.30608
## 6 15.85380
```

```r
#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_202308_TR_LysC_10cm   40.67031
## 2       11_MSM_202308_TR_LysB_10cm   18.75816
## 3      111_GWI_202308_WC_LysC_20cm   43.27226
## 4 121_MSM_202308_TR_RHZ_SF_Tree_7   24.73900
## 5       130_MSM_202308_WC_RHZ_LysC   31.00541
## 6       21_MSM_202308_WC_LysB_20cm   27.55851
```

```
#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_202308_TR_LysC_10cm 1119.1039  7253.308 34.90655 204.6067
## 2       11_MSM_202308_TR_LysB_10cm  512.6074  5592.842 15.98900 157.7671
## 3      111_GWI_202308_WC_LysC_20cm 1017.4247 11857.288 31.73502 334.4792
## 4 121_MSM_202308_TR_RHZ_SF_Tree_7  702.6218  7261.346 21.91584 204.8335
## 5       130_MSM_202308_WC_RHZ_LysC  830.9295  9806.238 25.91795 276.6217
## 6       21_MSM_202308_WC_LysB_20cm  736.3388  9223.311 22.96752 260.1780
##   salinity SO4_mM_spk
## 1 13.10675    40.67031
## 2 10.10629    18.75816
## 3 21.42614    43.27226
## 4 13.12128    24.73900
## 5 17.71990    31.00541
## 6 16.66655    27.55851
```

```
#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_202308_TR_LysC_10cm 1119.1039  7253.308 34.90655 204.6067
## 2       11_MSM_202308_TR_LysB_10cm  512.6074  5592.842 15.98900 157.7671
## 3      111_GWI_202308_WC_LysC_20cm 1017.4247 11857.288 31.73502 334.4792
## 4 121_MSM_202308_TR_RHZ_SF_Tree_7  702.6218  7261.346 21.91584 204.8335
## 5       130_MSM_202308_WC_RHZ_LysC  830.9295  9806.238 25.91795 276.6217
## 6       21_MSM_202308_WC_LysB_20cm  736.3388  9223.311 22.96752 260.1780
##   salinity SO4_mM_spk SO4_spk_Conc
## 1 13.10675    40.67031 7.797879e-05
## 2 10.10629    18.75816 7.797879e-05
## 3 21.42614    43.27226 7.797879e-05
## 4 13.12128    24.73900 7.797879e-05
## 5 17.71990    31.00541 7.797879e-05
## 6 16.66655    27.55851 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$Sample_ID, "MSM_202306_UP"), 50, QAspks$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)
```

```r
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_202308_TR_LysC_10cm 1119.1039  7253.308 34.90655 204.6067
## 2     11_MSM_202308_TR_LysB_10cm  512.6074  5592.842 15.98900 157.7671
## 3    111_GWI_202308_WC_LysC_20cm 1017.4247 11857.288 31.73502 334.4792
## 4 121_MSM_202308_TR_RHZ_SF_Tree_7  702.6218  7261.346 21.91584 204.8335
## 5      130_MSM_202308_WC_RHZ_LysC  830.9295  9806.238 25.91795 276.6217
## 6     21_MSM_202308_WC_LysB_20cm  736.3388  9223.311 22.96752 260.1780
##   salinity SO4_mM_spk SO4_spk_Conc Dilution SampleVol
## 1 13.10675   40.67031 7.797879e-05        1     1e-06
## 2 10.10629   18.75816 7.797879e-05        1     1e-06
## 3 21.42614   43.27226 7.797879e-05        1     1e-06
## 4 13.12128   24.73900 7.797879e-05        1     1e-06
## 5 17.71990   31.00541 7.797879e-05        1     1e-06
## 6 16.66655   27.55851 7.797879e-05        1     1e-06
```

```r
#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, rerur

head(QAspks)
```
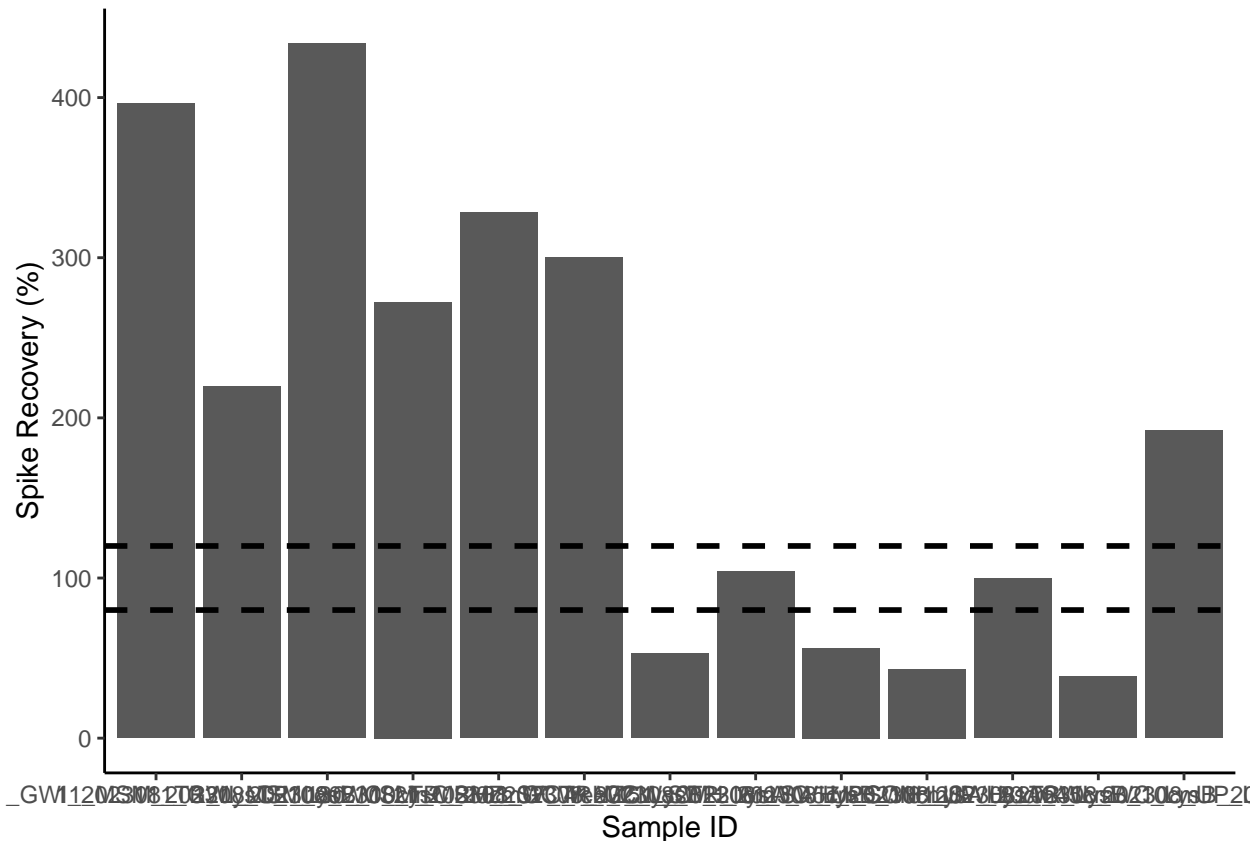
```
##                        Sample_ID   SO4_ppm    Cl_ppm    SO4_mM    Cl_mM
## 1     101_GWI_202308_TR_LysC_10cm 1119.1039  7253.308 34.90655 204.6067
## 2      11_MSM_202308_TR_LysB_10cm  512.6074  5592.842 15.98900 157.7671
## 3     111_GWI_202308_WC_LysC_20cm 1017.4247 11857.288 31.73502 334.4792
## 4 121_MSM_202308_TR_RHZ_SF_Tree_7  702.6218  7261.346 21.91584 204.8335
## 5      130_MSM_202308_WC_RHZ_LysC  830.9295  9806.238 25.91795 276.6217
## 6      21_MSM_202308_WC_LysB_20cm  736.3388  9223.311 22.96752 260.1780
##   salinity SO4_mM_spk SO4_spk_Conc Dilution SampleVol SO4_Total_unspkd
## 1 13.10675   40.67031 7.797879e-05        1     1e-06     3.490655e-05
## 2 10.10629   18.75816 7.797879e-05        1     1e-06     1.598900e-05
## 3 21.42614   43.27226 7.797879e-05        1     1e-06     3.173502e-05
## 4 13.12128   24.73900 7.797879e-05        1     1e-06     2.191584e-05
## 5 17.71990   31.00541 7.797879e-05        1     1e-06     2.591795e-05
## 6 16.66655   27.55851 7.797879e-05        1     1e-06     2.296752e-05
##   SO4_Total_spkd SO4_expctd_spkd spk_recovery SO4_spks_flag
## 1    0.0004473734    1.128853e-04     396.3078      NO, rerun
## 2    0.0002063398    9.396779e-05     219.5857      NO, rerun
## 3    0.0004759948    1.097138e-04     433.8513      NO, rerun
## 4    0.0002721290    9.989463e-05     272.4161      NO, rerun
## 5    0.0003410596    1.038967e-04     328.2678      NO, rerun
## 6    0.0003031436    1.009463e-04     300.3018      NO, rerun
```

```r
#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
```

```r
#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          11    13    84.6
## 2 YES                 2    13    15.4
```

Make final dataframe with IDs

```r
#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", "MSM", "202308", "UP", "LysA", "20cm"), c("10", :
## number of columns of result is not a multiple of vector length (arg 1)
```

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

```
##   Analysis_No Site   Date Zone Replicate Depth RHZ RHZ_Rep
## 1           1  MSM 202308   UP      LysA  20cm   1     MSM
## 2          10  MSM 202308   TR      LysA  45cm  10     MSM
## 3         100  GWI 202308   TR      LysB  45cm 100     GWI
## 4         101  GWI 202308   TR      LysC  10cm 101     GWI
## 5         102  GWI 202308   TR      LysC  20cm 102     GWI
## 6         103  GWI 202308   TR      LysC  45cm 103     GWI
```

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

```
##   Analysis_No Site   Date Zone Replicate Depth RHZ RHZ_Rep
## 1           1  MSM 202308   UP      LysA  20cm   1     MSM
## 2          10  MSM 202308   TR      LysA  45cm  10     MSM
## 3         100  GWI 202308   TR      LysB  45cm 100     GWI
## 4         101  GWI 202308   TR      LysC  10cm 101     GWI
## 5         102  GWI 202308   TR      LysC  20cm 102     GWI
## 6         103  GWI 202308   TR      LysC  45cm 103     GWI
##                   Sample_ID    SO4_ppm   Cl_ppm     SO4_mM      Cl_mM  salinity
## 1   1_MSM_202308_UP_LysA_20cm  149.8293 1634.498   4.673403   46.10715  2.953565
## 2  10_MSM_202308_TR_LysA_45cm  507.2198 4042.257  15.820954  114.02699  7.304384
## 3 100_GWI_202308_TR_LysB_45cm 1004.2827 8516.909  31.325100  240.25132 15.390081
## 4 101_GWI_202308_TR_LysC_10cm 1119.1039 7253.308  34.906547  204.60671 13.106753
## 5 102_GWI_202308_TR_LysC_20cm  892.7210 8024.734  27.845321  226.36766 14.500720
## 6 103_GWI_202308_TR_LysC_45cm 1093.3388 8828.044  34.102895  249.02805 15.952302
```

Make final dataframe with IDs

```r
write.csv(alldat, file="Processed Data/COMPASS_SynopticCB_PW_Processed_Cl_SO4_202308.csv")        #C
```

END