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_202309_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   n.a.     n.a.     n.a.     n.a.     n.a.
## 6 6 Standard 2 Calibration Standard 4.753   0.9994     6.87   1.1465     7.60
```

```
## 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     n.a.
## 6 Standard 2  0.9994
```

```
## 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      NA
## 6 Standard 2  0.9994
```

```
#Chloride data
## Read in raw data file from Dionex - copied and saved as a txt
Cldat <- read.table("Raw Data/COMPASS_Synoptic_CB_MonMon_202309_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 3.700  0.0065  100.00   0.0102    0.07
## 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 3.740  4.9411   93.38   7.7044   64.09
## 6 6 Standard 2 Calibration Standard 3.767  9.9736   93.13  15.5513  127.45
```

```
## 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 4.9411
## 6 Standard 2 9.9736
```

```
## 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 4.9411
## 6 Standard 2 9.9736
```

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

```
##   Sample_ID   SO4_ppm    Cl_ppm
## 1           1205.7288 9437.8836
## 2           1205.7288 2765.0461
## 3           1205.7288    0.0065
## 4           1205.7288 3281.0669
## 5           1205.7288        NA
## 6            279.7333 9437.8836
```

```
## 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_202309_UP_LysA_20cm   7.4516    3.3628
## 27     10_GCW_202309_WC_LysA_10cm 593.4642 6978.7764
## 28 10_GCW_202309_WC_LysA_10cm_dup 596.0266 7030.0612
## 29     11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026
## 30 11_GCW_202309_WC_LysA_20cm_spk 490.6059 6428.3445
## 31     12_GCW_202309_WC_LysA_45cm 228.8569 5506.8946
```

```
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_GCW_202309_UP_LysA_20cm   7.4516    3.3628
## 27     10_GCW_202309_WC_LysA_10cm 593.4642 6978.7764
## 28 10_GCW_202309_WC_LysA_10cm_dup 596.0266 7030.0612
## 29     11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026
## 30 11_GCW_202309_WC_LysA_20cm_spk 490.6059 6428.3445
## 31     12_GCW_202309_WC_LysA_45cm 228.8569 5506.8946
```

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

```
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
## 340 Standard 1  0.4898 5.0593
## 341 Standard 1  0.4898 5.2609
## 342 Standard 1  0.4898 4.9411
```

```
## 343 Standard 1  0.4898 5.1478
## 344 Standard 1  0.5051 5.0593
## 345 Standard 1  0.5051 5.2609
```

```
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  0.373  0.222  59.6  NO, rerun
## 2 Standard 2  1.03   0.0276  2.67 YES
## 3 Standard 3  2.04   0.0371  1.82 YES
## 4 Standard 4  7.11   5.33   75.0  NO, rerun
## 5 Standard 5 14.0   10.5    75.0  NO, rerun
```

```
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   5.10 0.121  2.37 YES
## 2 Standard 2  10.3  0.288  2.79 YES
## 3 Standard 3  20.0  0.432  2.16 YES
## 4 Standard 4 102.   2.18   2.13 YES
## 5 Standard 5 206.   4.37   2.12 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_202309_UP_LysA_20cm   7.4516    3.3628
## 27     10_GCW_202309_WC_LysA_10cm 593.4642 6978.7764
## 28 10_GCW_202309_WC_LysA_10cm_dup 596.0266 7030.0612
## 29     11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026
## 30 11_GCW_202309_WC_LysA_20cm_spk 490.6059 6428.3445
## 31     12_GCW_202309_WC_LysA_45cm 228.8569 5506.8946
```

```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_GCW_202309_UP_LysA_20cm   7.4516    3.3628  0.2324267   0.09486037
## 27     10_GCW_202309_WC_LysA_10cm 593.4642 6978.7764 18.5110480 196.86252186
## 28 10_GCW_202309_WC_LysA_10cm_dup 596.0266 7030.0612 18.5909732 198.30920169
## 29     11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026  9.4869994 185.31742172
## 30 11_GCW_202309_WC_LysA_20cm_spk 490.6059 6428.3445 15.3027417 181.33552891
## 31     12_GCW_202309_WC_LysA_45cm 228.8569 5506.8946  7.1383936 155.34258392
##       salinity
## 26  0.00610258
## 27 12.61067495
## 28 12.70334659
## 29 11.87111720
## 30 11.61604451
## 31  9.95098454
```

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_GCW_202309_UP_LysA_20cm   7.4516    3.3628  0.2324267   0.09486037
## 27     10_GCW_202309_WC_LysA_10cm 593.4642 6978.7764 18.5110480 196.86252186
## 28 10_GCW_202309_WC_LysA_10cm_dup 596.0266 7030.0612 18.5909732 198.30920169
## 29     11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026  9.4869994 185.31742172
## 30 11_GCW_202309_WC_LysA_20cm_spk 490.6059 6428.3445 15.3027417 181.33552891
## 31     12_GCW_202309_WC_LysA_45cm 228.8569 5506.8946  7.1383936 155.34258392
##       salinity
## 26  0.00610258
## 27 12.61067495
## 28 12.70334659
```

```
## 29 11.87111720
## 30 11.61604451
## 31  9.95098454
```

```
#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_202309_WC_LysA_10cm_dup 596.0266 7030.0612 18.5909732 1.983092e+02
## 2           20_GCW_202309_SW_B_dup 770.0896 5898.2462 24.0202620 1.663821e+02
## 3    30_MSM_202309_TR_LysA_45cm_dup 251.9186 1988.5976  7.8577230 5.609584e+01
## 4    40_MSM_202309_WC_LysB_10cm_dup 634.3614 8701.6746 19.7866937 2.454633e+02
## 5 50_SWH_202309_UPCON_LysA_20cm_dup   0.0000    0.0679  0.0000000 1.915374e-03
## 6    60_SWH_202309_UP_LysA_45cm_dup   5.4856  214.0801  0.1711042 6.038931e+00
##       salinity
## 1 1.270335e+01
## 2 1.065816e+01
## 3 3.593422e+00
## 4 1.572395e+01
## 5 1.486953e-04
## 6 3.868687e-01
```

```
#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_202309_UP_LysA_20cm   7.4516    3.3628  0.2324267   0.09486037
## 2 10_GCW_202309_WC_LysA_10cm 593.4642 6978.7764 18.5110480 196.86252186
## 3 11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026  9.4869994 185.31742172
## 4 12_GCW_202309_WC_LysA_45cm 228.8569 5506.8946  7.1383936 155.34258392
## 5 13_GCW_202309_WC_LysB_10cm 293.2979 6840.1267  9.1484061 192.95138787
## 6 14_GCW_202309_WC_LysB_20cm 134.6521 6335.1128  4.2000031 178.70557969
##      salinity
## 1  0.00610258
## 2 12.61067495
## 3 11.87111720
## 4  9.95098454
## 5 12.36013495
## 6 11.44757483
```

```
#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_202309_WC_LysA_10cm 18.5909732 1.983092e+02 1.270335e+01
## 2            20_GCW_202309_SW_B 24.0202620 1.663821e+02 1.065816e+01
## 3    30_MSM_202309_TR_LysA_45cm  7.8577230 5.609584e+01 3.593422e+00
## 4    40_MSM_202309_WC_LysB_10cm 19.7866937 2.454633e+02 1.572395e+01
## 5 50_SWH_202309_UPCON_LysA_20cm  0.0000000 1.915374e-03 1.486953e-04
## 6    60_SWH_202309_UP_LysA_45cm  0.1711042 6.038931e+00 3.868687e-01
```

```
#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_202309_WC_LysA_10cm 593.4642 6978.7764 18.51104803 196.86252186
## 2            20_GCW_202309_SW_B 809.5272 6202.5891 25.25038054 174.96725247
## 3    30_MSM_202309_TR_LysA_45cm 503.5731 3988.6796 15.70720836 112.51564457
## 4    40_MSM_202309_WC_LysB_10cm 685.5478 9424.2428 21.38327511 265.84605924
## 5 50_SWH_202309_UPCON_LysA_20cm   0.1084    0.6823  0.00338116   0.01924683
## 6    60_SWH_202309_UP_LysA_45cm   7.1839  227.6073  0.22407673   6.42051622
##       salinity SO4_mM_dup    Cl_mM_dup salinity_dup
## 1 12.610674955 18.5909732 1.983092e+02 1.270335e+01
## 2 11.208104504 24.0202620 1.663821e+02 1.065816e+01
## 3  7.207570037  7.8577230 5.609584e+01 3.593422e+00
## 4 17.029632740 19.7866937 2.454633e+02 1.572395e+01
## 5  0.001258916  0.0000000 1.915374e-03 1.486953e-04
## 6  0.411312391  0.1711042 6.038931e+00 3.868687e-01
```

```
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)
```

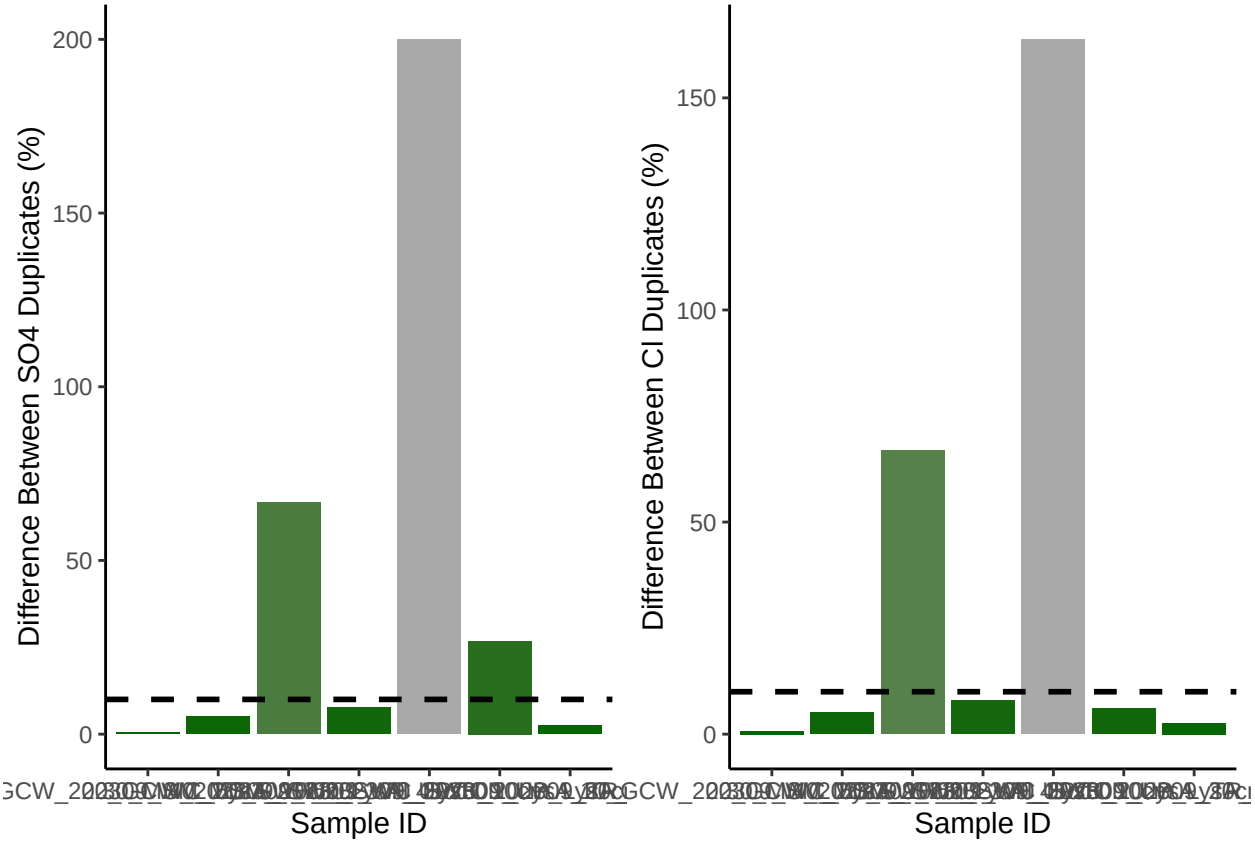
```
##                       Sample_ID  SO4_ppm    Cl_ppm      SO4_mM        Cl_mM
## 1    10_GCW_202309_WC_LysA_10cm 593.4642 6978.7764 18.51104803 196.86252186
## 2            20_GCW_202309_SW_B 809.5272 6202.5891 25.25038054 174.96725247
## 3    30_MSM_202309_TR_LysA_45cm 503.5731 3988.6796 15.70720836 112.51564457
## 4    40_MSM_202309_WC_LysB_10cm 685.5478 9424.2428 21.38327511 265.84605924
## 5 50_SWH_202309_UPCON_LysA_20cm   0.1084    0.6823  0.00338116   0.01924683
## 6    60_SWH_202309_UP_LysA_45cm   7.1839  227.6073  0.22407673   6.42051622
##       salinity SO4_mM_dup    Cl_mM_dup salinity_dup SO4_dups_chk SO4_dups_flag
## 1 12.610674955 18.5909732 1.983092e+02 1.270335e+01    0.4308398           YES
## 2 11.208104504 24.0202620 1.663821e+02 1.065816e+01    4.9933123           YES
## 3  7.207570037  7.8577230 5.609584e+01 3.593422e+00   66.6200568     NO, rerun
## 4 17.029632740 19.7866937 2.454633e+02 1.572395e+01    7.7560487           YES
## 5  0.001258916  0.0000000 1.915374e-03 1.486953e-04  200.0000000     NO, rerun
## 6  0.411312391  0.1711042 6.038931e+00 3.868687e-01   26.8092663     NO, rerun
##   Cl_dups_chk Cl_dups_flag
## 1   0.7321778           YES
## 2   5.0301139           YES
## 3  66.9228457    NO, rerun
## 4   7.9727628           YES
```

```
## 5 163.7963210    NO, rerun
## 6   6.1252370          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)
```



```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         3 NO, rerun          2     7  42.85714   28.57143
## 2       YES         4       YES          5     7  57.14286   71.42857
```

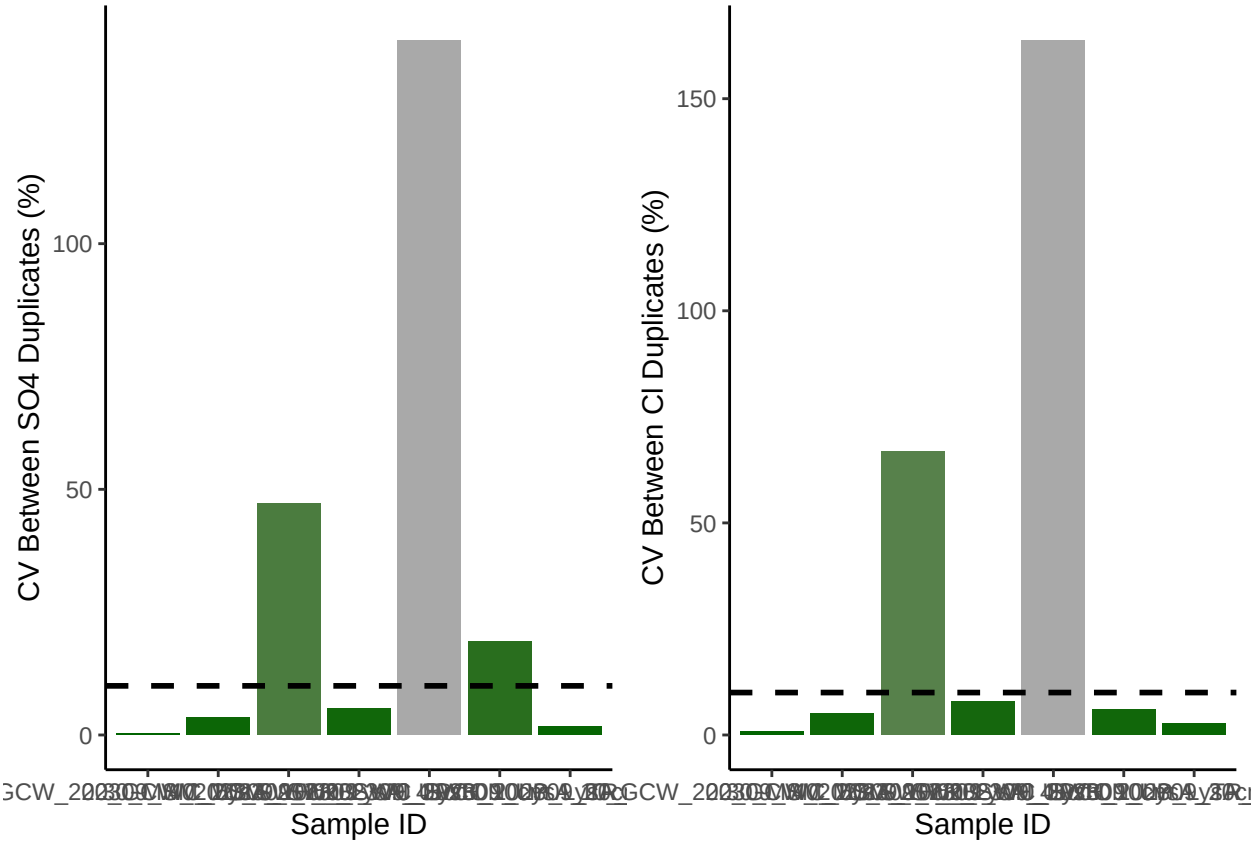
Pull out dups and check with cv

```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_GCW_202309_WC_LysA_10cm 593.4642 6978.7764 18.51104803 196.86252186
## 2            20_GCW_202309_SW_B 809.5272 6202.5891 25.25038054 174.96725247
## 3    30_MSM_202309_TR_LysA_45cm 503.5731 3988.6796 15.70720836 112.51564457
## 4    40_MSM_202309_WC_LysB_10cm 685.5478 9424.2428 21.38327511 265.84605924
## 5 50_SWH_202309_UPCON_LysA_20cm   0.1084    0.6823  0.00338116   0.01924683
## 6    60_SWH_202309_UP_LysA_45cm   7.1839  227.6073  0.22407673   6.42051622
##       salinity SO4_mM_dup    Cl_mM_dup salinity_dup SO4_dups_chk SO4_dups_flag
## 1 12.610674955 18.5909732 1.983092e+02 1.270335e+01    0.4308398           YES
## 2 11.208104504 24.0202620 1.663821e+02 1.065816e+01    4.9933123           YES
## 3  7.207570037  7.8577230 5.609584e+01 3.593422e+00   66.6200568     NO, rerun
## 4 17.029632740 19.7866937 2.454633e+02 1.572395e+01    7.7560487           YES
## 5  0.001258916  0.0000000 1.915374e-03 1.486953e-04  200.0000000     NO, rerun
## 6  0.411312391  0.1711042 6.038931e+00 3.868687e-01   26.8092663     NO, rerun
##   Cl_dups_chk Cl_dups_flag SO4_dups_cv SO4_dups_cv_flag
## 1   0.7321778          YES   0.3046498              YES
```

```
## 2   5.0301139          YES   3.5308050             YES
## 3  66.9228457    NO, rerun  47.1074940       NO, rerun
## 4   7.9727628          YES   5.4843546             YES
## 5 163.7963210    NO, rerun 141.4213562       NO, rerun
## 6   6.1252370          YES  18.9570140       NO, rerun
```

```
#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 NO, rerun         3 NO, rerun          2     7  42.85714   28.57143
## 2       YES         4       YES          5     7  57.14286   71.42857
```

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_202309_UP_LysA_20cm   7.4516    3.3628  0.2324267   0.09486037
## 27     10_GCW_202309_WC_LysA_10cm 593.4642 6978.7764 18.5110480 196.86252186
## 28 10_GCW_202309_WC_LysA_10cm_dup 596.0266 7030.0612 18.5909732 198.30920169
## 29     11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026  9.4869994 185.31742172
## 30 11_GCW_202309_WC_LysA_20cm_spk 490.6059 6428.3445 15.3027417 181.33552891
## 31     12_GCW_202309_WC_LysA_45cm 228.8569 5506.8946  7.1383936 155.34258392
##       salinity
## 26  0.00610258
## 27 12.61067495
## 28 12.70334659
## 29 11.87111720
## 30 11.61604451
## 31  9.95098454
```

```
#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    11_GCW_202309_WC_LysA_20cm_spk 490.6059 6428.3445 15.3027417 181.335529
## 2            21_GCW_202309_SW_C_spk 958.1205 5884.6478 29.8852308 165.998528
## 3    31_MSM_202309_TR_LysB_10cm_spk 444.0675 3520.5436 13.8511385  99.310116
## 4    41_MSM_202309_WC_LysB_20cm_spk 922.5773 9076.6365 28.7765845 256.040522
```

```
## 5 51_SWH_202309_UPCON_LysA_45cm_spk 138.4937  108.5239  4.3198284   3.061323
## 6    61_SWH_202309_UP_LysB_10cm_spk   5.5928  165.5009  0.1744479   4.668573
##     salinity
## 1 11.6160445
## 2 10.6335846
## 3  6.3616483
## 4 16.4015082
## 5  0.1961287
## 6  0.2990861
```

```
#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    11_GCW_202309_WC_LysA_20cm 15.3027417
## 2            21_GCW_202309_SW_C 29.8852308
## 3    31_MSM_202309_TR_LysB_10cm 13.8511385
## 4    41_MSM_202309_WC_LysB_20cm 28.7765845
## 5 51_SWH_202309_UPCON_LysA_45cm  4.3198284
## 6    61_SWH_202309_UP_LysB_10cm  0.1744479
```

```
#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    11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026  9.4869994 185.317422
## 2            21_GCW_202309_SW_C 811.2022 6193.2976 25.3026263 174.705151
## 3    31_MSM_202309_TR_LysB_10cm 675.2835 6882.4336 21.0631160 194.144812
## 4    41_MSM_202309_WC_LysB_20cm 728.9486 9246.8867 22.7370119 260.843066
## 5 51_SWH_202309_UPCON_LysA_45cm  30.3570  117.3651  0.9468808   3.310722
## 6    61_SWH_202309_UP_LysB_10cm  48.4169  502.6916  1.5101965  14.180299
##     salinity SO4_mM_spk
## 1 11.8711172 15.3027417
## 2 11.1913148 29.8852308
## 3 12.4365835 13.8511385
## 4 16.7091503 28.7765845
## 5  0.2121047  4.3198284
## 6  0.9083897  0.1744479
```

```
#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    11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026  9.4869994 185.317422
## 2            21_GCW_202309_SW_C 811.2022 6193.2976 25.3026263 174.705151
## 3    31_MSM_202309_TR_LysB_10cm 675.2835 6882.4336 21.0631160 194.144812
## 4    41_MSM_202309_WC_LysB_20cm 728.9486 9246.8867 22.7370119 260.843066
## 5 51_SWH_202309_UPCON_LysA_45cm  30.3570  117.3651  0.9468808   3.310722
## 6    61_SWH_202309_UP_LysB_10cm  48.4169  502.6916  1.5101965  14.180299
##     salinity SO4_mM_spk SO4_spk_Conc
## 1 11.8711172 15.3027417 7.797879e-05
## 2 11.1913148 29.8852308 7.797879e-05
## 3 12.4365835 13.8511385 7.797879e-05
## 4 16.7091503 28.7765845 7.797879e-05
## 5  0.2121047  4.3198284 7.797879e-05
## 6  0.9083897  0.1744479 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$Dilution)
QAspks$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    11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026  9.4869994 185.317422
## 2            21_GCW_202309_SW_C 811.2022 6193.2976 25.3026263 174.705151
## 3    31_MSM_202309_TR_LysB_10cm 675.2835 6882.4336 21.0631160 194.144812
```

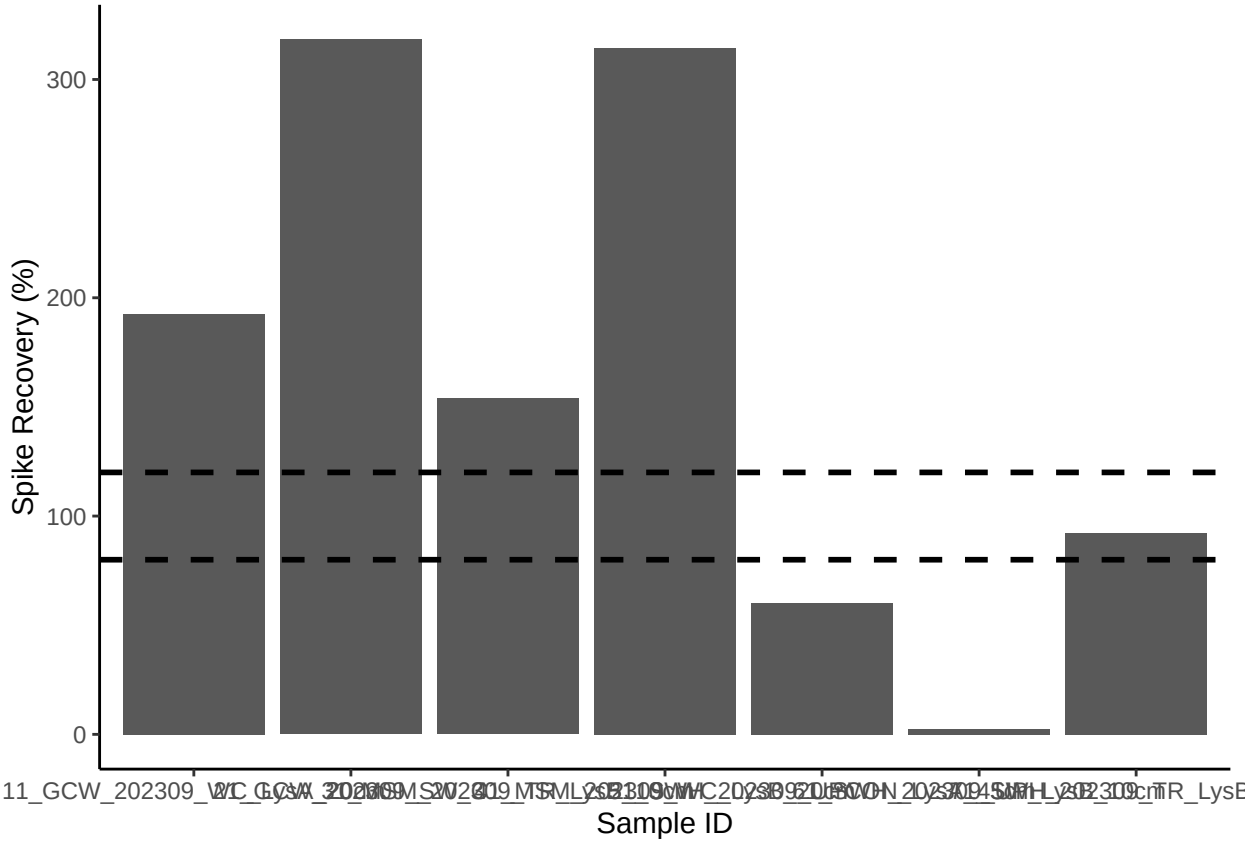
```
## 4    41_MSM_202309_WC_LysB_20cm 728.9486 9246.8867 22.7370119 260.843066
## 5 51_SWH_202309_UPCON_LysA_45cm  30.3570  117.3651  0.9468808   3.310722
## 6    61_SWH_202309_UP_LysB_10cm  48.4169  502.6916  1.5101965  14.180299
##     salinity SO4_mM_spk SO4_spk_Conc Dilution SampleVol
## 1 11.8711172 15.3027417 7.797879e-05        1     1e-06
## 2 11.1913148 29.8852308 7.797879e-05        1     1e-06
## 3 12.4365835 13.8511385 7.797879e-05        1     1e-06
## 4 16.7091503 28.7765845 7.797879e-05        1     1e-06
## 5  0.2121047  4.3198284 7.797879e-05        1     1e-06
## 6  0.9083897  0.1744479 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
## 1    11_GCW_202309_WC_LysA_20cm 304.1532 6569.5026  9.4869994 185.317422
## 2            21_GCW_202309_SW_C 811.2022 6193.2976 25.3026263 174.705151
## 3    31_MSM_202309_TR_LysB_10cm 675.2835 6882.4336 21.0631160 194.144812
## 4    41_MSM_202309_WC_LysB_20cm 728.9486 9246.8867 22.7370119 260.843066
## 5 51_SWH_202309_UPCON_LysA_45cm  30.3570  117.3651  0.9468808   3.310722
## 6    61_SWH_202309_UP_LysB_10cm  48.4169  502.6916  1.5101965  14.180299
##     salinity SO4_mM_spk SO4_spk_Conc Dilution SampleVol SO4_Total_unspkd
## 1 11.8711172 15.3027417 7.797879e-05        1     1e-06     9.486999e-06
## 2 11.1913148 29.8852308 7.797879e-05        1     1e-06     2.530263e-05
## 3 12.4365835 13.8511385 7.797879e-05        1     1e-06     2.106312e-05
## 4 16.7091503 28.7765845 7.797879e-05        1     1e-06     2.273701e-05
## 5  0.2121047  4.3198284 7.797879e-05        1     1e-06     9.468808e-07
## 6  0.9083897  0.1744479 7.797879e-05        1     1e-06     1.510197e-06
##   SO4_Total_spkd SO4_expctd_spkd spk_recovery SO4_spks_flag
## 1   1.683302e-04    8.746579e-05   192.452570     NO, rerun
## 2   3.287375e-04    1.032814e-04   318.293021     NO, rerun
## 3   1.523625e-04    9.904191e-05   153.836421     NO, rerun
## 4   3.165424e-04    1.007158e-04   314.292717     NO, rerun
## 5   4.751811e-05    7.892567e-05    60.206157     NO, rerun
## 6   1.918927e-06    7.948899e-05     2.414079     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: 2 x 4
##   SO4_spks_flag no_rows Total Percent
##   <chr>           <int> <int>   <dbl>
## 1 NO, rerun           6     7    85.7
## 2 YES                 1     7    14.3
```

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", "202309", "UP", "LysA", "20cm"), c("10", :
## number of columns of result is not a multiple of vector length (arg 11)
```

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

```
##   Analysis_No Site   Date Zone Replicate Depth
## 1           1  GCW 202309   UP      LysA  20cm
## 2          10  GCW 202309   WC      LysA  10cm
## 3          11  GCW 202309   WC      LysA  20cm
## 4          12  GCW 202309   WC      LysA  45cm
## 5          13  GCW 202309   WC      LysB  10cm
## 6          14  GCW 202309   WC      LysB  20cm
```

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

```
##   Analysis_No Site   Date Zone Replicate Depth                  Sample_ID
## 1           1  GCW 202309   UP      LysA  20cm  1_GCW_202309_UP_LysA_20cm
## 2          10  GCW 202309   WC      LysA  10cm 10_GCW_202309_WC_LysA_10cm
## 3          11  GCW 202309   WC      LysA  20cm 11_GCW_202309_WC_LysA_20cm
## 4          12  GCW 202309   WC      LysA  45cm 12_GCW_202309_WC_LysA_45cm
## 5          13  GCW 202309   WC      LysB  10cm 13_GCW_202309_WC_LysB_10cm
## 6          14  GCW 202309   WC      LysB  20cm 14_GCW_202309_WC_LysB_20cm
##    SO4_ppm    Cl_ppm     SO4_mM        Cl_mM    salinity
## 1   7.4516    3.3628  0.2324267   0.09486037  0.00610258
## 2 593.4642 6978.7764 18.5110480 196.86252186 12.61067495
## 3 304.1532 6569.5026  9.4869994 185.31742172 11.87111720
## 4 228.8569 5506.8946  7.1383936 155.34258392  9.95098454
## 5 293.2979 6840.1267  9.1484061 192.95138787 12.36013495
## 6 134.6521 6335.1128  4.2000031 178.70557969 11.44757483
```

Make final dataframe with IDs

```
write.csv(alldat, file="Processed Data/COMPASS_SynopticCB_PW_Processed_Cl_SO4_202309.csv")           #C
```

END