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_202307_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 4.787   0.0102     1.22   0.0126     0.06
## 2 2  Lab Blank              Unknown 4.463   0.0113     1.39   0.0139     0.01
## 3 3  Lab Blank              Unknown   n.a.     n.a.     n.a.     n.a.     n.a.
## 4 4  Lab Blank              Unknown 4.793   0.0082     1.00   0.0102     0.04
## 5 5 Standard 1 Calibration Standard 4.827   0.4524     6.61   0.5582     3.71
## 6 6 Standard 2 Calibration Standard 4.797   0.9445     6.83   1.1653     7.83
```

```
## 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  0.0102
## 2  Lab Blank  0.0113
## 3  Lab Blank     n.a.
## 4  Lab Blank  0.0082
## 5 Standard 1  0.4524
## 6 Standard 2  0.9445
```

```
## 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  0.0102
## 2  Lab Blank  0.0113
## 3  Lab Blank      NA
## 4  Lab Blank  0.0082
## 5 Standard 1  0.4524
## 6 Standard 2  0.9445
```

```
#Chloride data
## Read in raw data file from Dionex - copied and saved as a txt
Cldat <- read.table("Raw Data/COMPASS_Synoptic_CB_MonMon_202307_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 3.813  0.0329    5.17   0.0536    0.33
## 2 2  Lab Blank             Unknown 3.827  0.0380    6.18   0.0618    0.39
## 3 3  Lab Blank             Unknown 3.793  0.0414    6.76   0.0674    0.53
## 4 4  Lab Blank             Unknown 3.827  0.0448    7.17   0.0729    0.58
## 5 5 Standard 1 Calibration Standard 3.800  4.8395   93.39   7.8835   64.81
## 6 6 Standard 2 Calibration Standard 3.827  9.7556   93.17  15.8916  130.12
```

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

```
##          X.1 IC.Cl.1
## 1  Lab Blank 0.0329
## 2  Lab Blank 0.0380
## 3  Lab Blank 0.0414
## 4  Lab Blank 0.0448
## 5 Standard 1 4.8395
## 6 Standard 2 9.7556
```

```
## 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 0.0329
## 2  Lab Blank 0.0380
## 3  Lab Blank 0.0414
## 4  Lab Blank 0.0448
## 5 Standard 1 4.8395
## 6 Standard 2 9.7556
```

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

```
##   Sample_ID   SO4_ppm     Cl_ppm
## 1           1580.7081 12144.2634
## 2           1580.7081  3641.7012
## 3           1580.7081     0.0164
## 4           1580.7081  3944.6183
## 5           1580.7081         NA
## 6            398.0446 12144.2634
```

```
## 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_202307_UP_LysA_20cm  8.3626   1.2711
## 27      10_GCW_202307_TR_LysC_20cm  6.5457  62.1675
## 28  10_GCW_202307_TR_LysC_20cm_dup  7.2180  63.0616
## 29     100_SWH_202307_TR_LysB_10cm 21.1419 621.7679
## 30 100_SWH_202307_TR_LysB_10cm_dup 21.4371 628.7818
## 31     101_SWH_202307_TR_LysB_20cm      NA 640.1021
```

```
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_202307_UP_LysA_20cm  8.3626   1.2711
## 27      10_GCW_202307_TR_LysC_20cm  6.5457  62.1675
## 28  10_GCW_202307_TR_LysC_20cm_dup  7.2180  63.0616
## 29     100_SWH_202307_TR_LysB_10cm 21.1419 621.7679
## 30 100_SWH_202307_TR_LysB_10cm_dup 21.4371 628.7818
## 31     101_SWH_202307_TR_LysB_20cm  0.0000 640.1021
```

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
## 585 Standard 1  0.4487 4.8366
## 586 Standard 1  0.4487 4.7322
## 587 Standard 1  0.4487 4.8724
```

```
## 588 Standard 1  0.4487 4.7467
## 589 Standard 1  0.4487 4.8395
## 590 Standard 1  0.4377 4.8366
```

```
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  0.448 0.00864  1.93 YES
## 2 Standard 2  0.935 0.00941  1.01 YES
## 3 Standard 3  1.87  0.0443   2.36 YES
## 4 Standard 4  9.56  0.135    1.41 YES
## 5 Standard 5 20.0   0.219    1.10 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   4.81 0.0558 1.16  YES
## 2 Standard 2   9.64 0.0889 0.922 YES
## 3 Standard 3  18.9  0.210  1.11  YES
## 4 Standard 4  95.6  1.48   1.55  YES
## 5 Standard 5 200.   2.12   1.06  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_202307_UP_LysA_20cm  8.3626   1.2711
## 27      10_GCW_202307_TR_LysC_20cm  6.5457  62.1675
## 28  10_GCW_202307_TR_LysC_20cm_dup  7.2180  63.0616
## 29     100_SWH_202307_TR_LysB_10cm 21.1419 621.7679
## 30 100_SWH_202307_TR_LysB_10cm_dup 21.4371 628.7818
## 31     101_SWH_202307_TR_LysB_20cm  0.0000 640.1021
```

```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_202307_UP_LysA_20cm  8.3626   1.2711 0.2608422  0.03585614
## 27      10_GCW_202307_TR_LysC_20cm  6.5457  62.1675 0.2041703  1.75366714
## 28  10_GCW_202307_TR_LysC_20cm_dup  7.2180  63.0616 0.2251404  1.77888858
## 29     100_SWH_202307_TR_LysB_10cm 21.1419 621.7679 0.6594479 17.53929196
## 30 100_SWH_202307_TR_LysB_10cm_dup 21.4371 628.7818 0.6686556 17.73714528
## 31     101_SWH_202307_TR_LysB_20cm  0.0000 640.1021 0.0000000 18.05647673
##       salinity
## 26 0.002322878
## 27 0.112362672
## 28 0.113978311
## 29 1.123560595
## 30 1.136234713
## 31 1.156690495
```

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_202307_UP_LysA_20cm  8.3626   1.2711 0.2608422  0.03585614
## 27      10_GCW_202307_TR_LysC_20cm  6.5457  62.1675 0.2041703  1.75366714
## 28  10_GCW_202307_TR_LysC_20cm_dup  7.2180  63.0616 0.2251404  1.77888858
## 29     100_SWH_202307_TR_LysB_10cm 21.1419 621.7679 0.6594479 17.53929196
## 30 100_SWH_202307_TR_LysB_10cm_dup 21.4371 628.7818 0.6686556 17.73714528
## 31     101_SWH_202307_TR_LysB_20cm  0.0000 640.1021 0.0000000 18.05647673
##       salinity
## 26 0.002322878
## 27 0.112362672
## 28 0.113978311
```

```
## 29 1.123560595
## 30 1.136234713
## 31 1.156690495
```

```
#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_202307_TR_LysC_20cm_dup   7.2180   63.0616  0.22514036   1.778889
## 2     100_SWH_202307_TR_LysB_10cm_dup  21.4371  628.7818  0.66865565  17.737145
## 3     110_SWH_202307_WC_LysB_20cm_dup   0.5585  933.0750  0.01742046  26.320874
## 4 120_MSM_202307_TR_RHZ_SF_Tree_3_dup 887.8881 7858.3687 27.69457580 221.674717
## 5  130_MSM_202307_WC_RHZ_Collar_2_dup 855.2333 8705.0128 26.67602308 245.557484
## 6      20_GCW_202307_WC_LysC_45cm_dup 516.6947 5052.6198 16.11649095 142.528062
##     salinity
## 1  0.1139783
## 2  1.1362347
## 3  1.6860925
## 4 14.2000982
## 5 15.7299841
## 6  9.1301100
```

```
#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_202307_UP_LysA_20cm  8.3626   1.2711 0.2608422  0.03585614
## 2  10_GCW_202307_TR_LysC_20cm  6.5457  62.1675 0.2041703  1.75366714
## 3 100_SWH_202307_TR_LysB_10cm 21.1419 621.7679 0.6594479 17.53929196
## 4 101_SWH_202307_TR_LysB_20cm  0.0000 640.1021 0.0000000 18.05647673
## 5 102_SWH_202307_TR_LysB_45cm  0.0000 791.2957 0.0000000 22.32145839
## 6 103_SWH_202307_TR_LysC_10cm 46.0312 732.7850 1.4357829 20.67094499
##      salinity
## 1 0.002322878
## 2 0.112362672
## 3 1.123560595
## 4 1.156690495
## 5 1.429897330
## 6 1.324168495
```

```
#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_202307_TR_LysC_20cm  0.22514036   1.778889    0.1139783
## 2     100_SWH_202307_TR_LysB_10cm  0.66865565  17.737145    1.1362347
## 3     110_SWH_202307_WC_LysB_20cm  0.01742046  26.320874    1.6860925
## 4 120_MSM_202307_TR_RHZ_SF_Tree_3 27.69457580 221.674717   14.2000982
## 5  130_MSM_202307_WC_RHZ_Collar_2 26.67602308 245.557484   15.7299841
## 6      20_GCW_202307_WC_LysC_45cm 16.11649095 142.528062    9.1301100
```

```
#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_202307_TR_LysC_20cm   6.5457   62.1675  0.20417031   1.753667
## 2     100_SWH_202307_TR_LysB_10cm  21.1419  621.7679  0.65944791  17.539292
## 3     110_SWH_202307_WC_LysB_20cm   0.5574  923.2323  0.01738615  26.043224
## 4 120_MSM_202307_TR_RHZ_SF_Tree_3 871.2304 7708.1261 27.17499688 217.436561
## 5  130_MSM_202307_WC_RHZ_Collar_2 848.7217 8630.1204 26.47291641 243.444863
## 6      20_GCW_202307_WC_LysC_45cm 513.9643 5010.2501 16.03132564 141.332866
##     salinity  SO4_mM_dup  Cl_mM_dup salinity_dup
## 1  0.1123627  0.22514036   1.778889    0.1139783
## 2  1.1235606  0.66865565  17.737145    1.1362347
## 3  1.6683068  0.01742046  26.320874    1.6860925
## 4 13.9286099 27.69457580 221.674717   14.2000982
## 5 15.5946536 26.67602308 245.557484   15.7299841
## 6  9.0535479 16.11649095 142.528062    9.1301100
```

```
QAdups$SO4_dups_chk <- ((abs(QAdups$SO4_mM-QAdups$SO4_mM_dup))/((QAdups$SO4_mM+QAdups$SO4_mM_dup)/2))*10
QAdups$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)
```

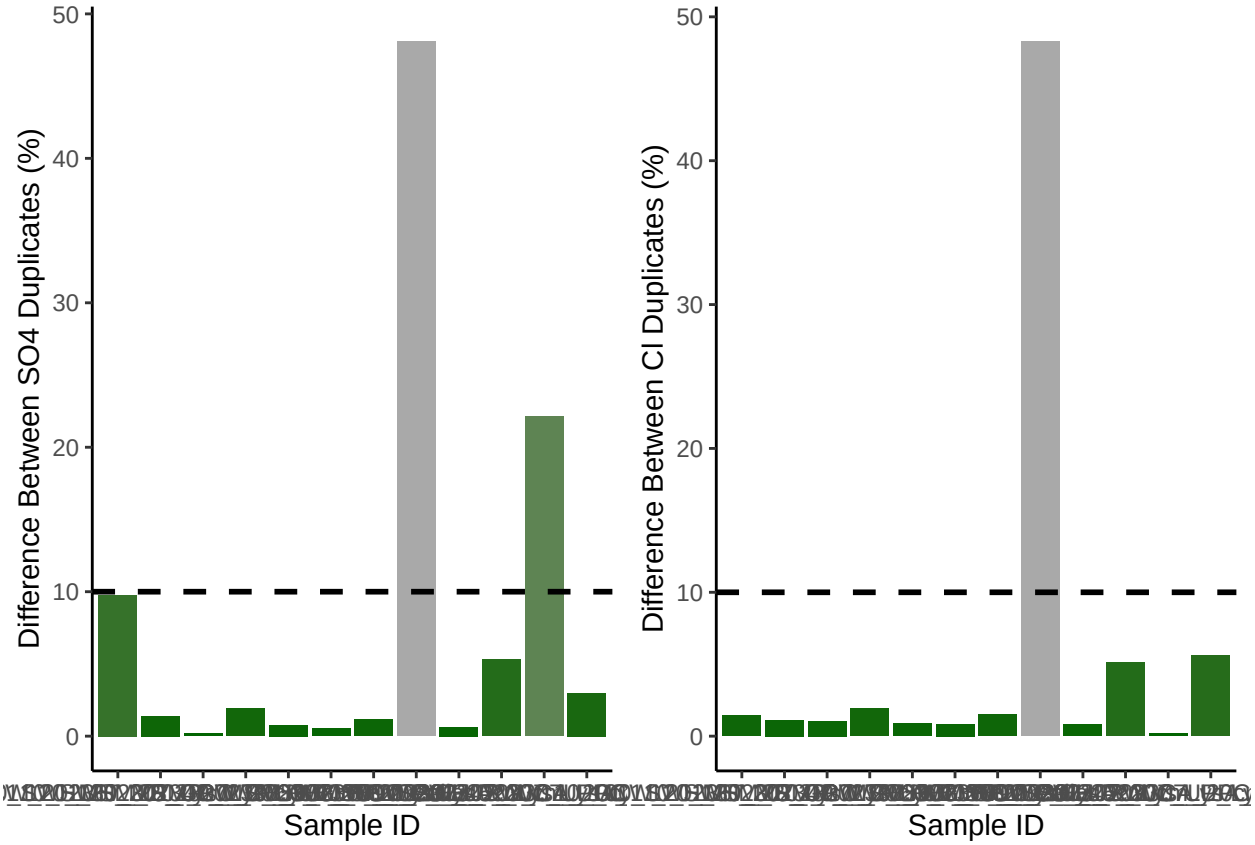
```
##                         Sample_ID  SO4_ppm    Cl_ppm      SO4_mM      Cl_mM
## 1      10_GCW_202307_TR_LysC_20cm   6.5457   62.1675  0.20417031   1.753667
## 2     100_SWH_202307_TR_LysB_10cm  21.1419  621.7679  0.65944791  17.539292
## 3     110_SWH_202307_WC_LysB_20cm   0.5574  923.2323  0.01738615  26.043224
## 4 120_MSM_202307_TR_RHZ_SF_Tree_3 871.2304 7708.1261 27.17499688 217.436561
## 5  130_MSM_202307_WC_RHZ_Collar_2 848.7217 8630.1204 26.47291641 243.444863
## 6      20_GCW_202307_WC_LysC_45cm 513.9643 5010.2501 16.03132564 141.332866
##     salinity  SO4_mM_dup  Cl_mM_dup salinity_dup SO4_dups_chk SO4_dups_flag
## 1  0.1123627  0.22514036   1.778889    0.1139783    9.7691754           YES
## 2  1.1235606  0.66865565  17.737145    1.1362347    1.3865990           YES
## 3  1.6683068  0.01742046  26.320874    1.6860925    0.1971503           YES
## 4 13.9286099 27.69457580 221.674717   14.2000982    1.8938690           YES
## 5 15.5946536 26.67602308 245.557484   15.7299841    0.7642925           YES
## 6  9.0535479 16.11649095 142.528062    9.1301100    0.5298358           YES
##   Cl_dups_chk Cl_dups_flag
## 1   1.4279429          YES
## 2   1.1217307          YES
## 3   1.0604602          YES
## 4   1.9303331          YES
```

```
## 5   0.8640534          YES
## 6   0.8420997          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         2 NO, rerun          1    12  16.66667   8.333333
## 2       YES        10       YES         11    12  83.33333  91.666667
```

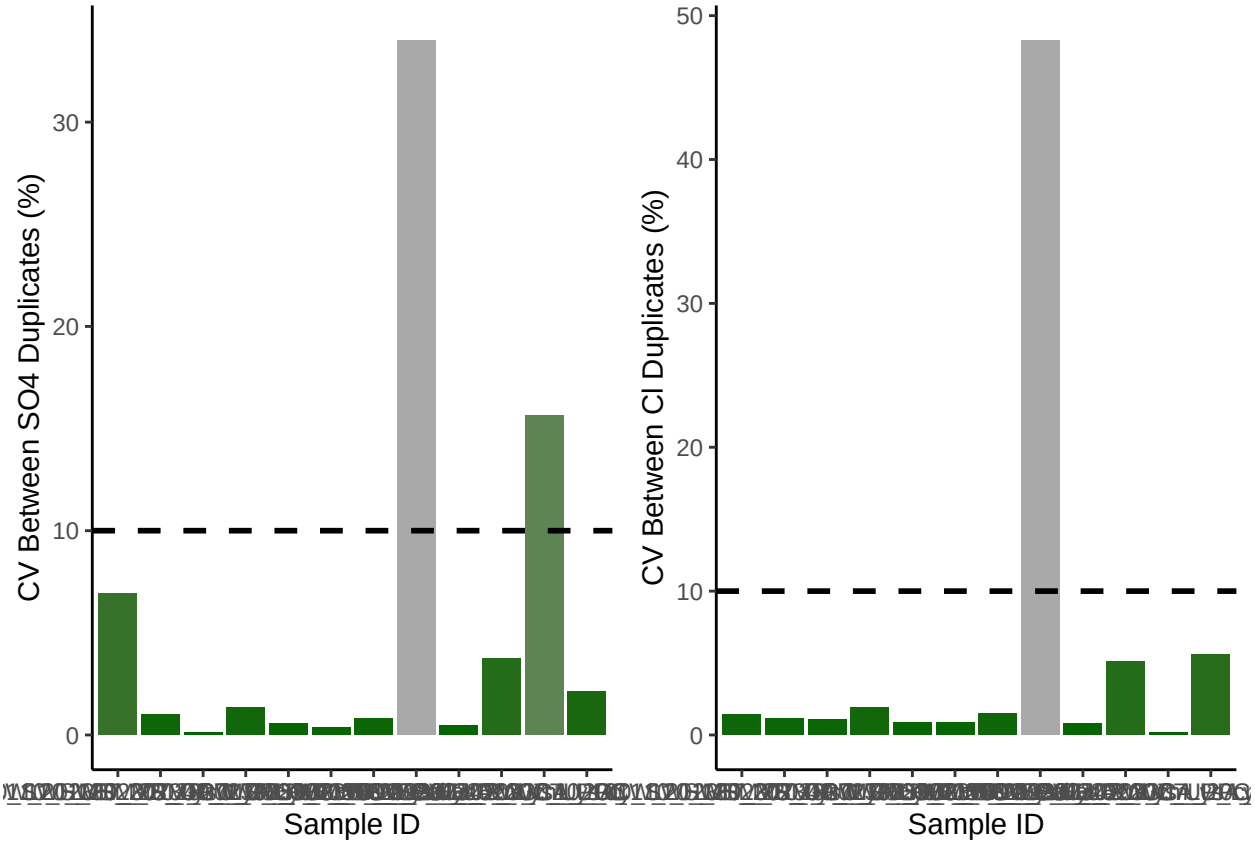
Pull out dups and check with cv

```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_202307_TR_LysC_20cm   6.5457   62.1675  0.20417031   1.753667
## 2     100_SWH_202307_TR_LysB_10cm  21.1419  621.7679  0.65944791  17.539292
## 3     110_SWH_202307_WC_LysB_20cm   0.5574  923.2323  0.01738615  26.043224
## 4 120_MSM_202307_TR_RHZ_SF_Tree_3 871.2304 7708.1261 27.17499688 217.436561
## 5  130_MSM_202307_WC_RHZ_Collar_2 848.7217 8630.1204 26.47291641 243.444863
## 6      20_GCW_202307_WC_LysC_45cm 513.9643 5010.2501 16.03132564 141.332866
##     salinity  SO4_mM_dup  Cl_mM_dup salinity_dup SO4_dups_chk SO4_dups_flag
## 1  0.1123627  0.22514036   1.778889    0.1139783    9.7691754           YES
## 2  1.1235606  0.66865565  17.737145    1.1362347    1.3865990           YES
## 3  1.6683068  0.01742046  26.320874    1.6860925    0.1971503           YES
## 4 13.9286099 27.69457580 221.674717   14.2000982    1.8938690           YES
## 5 15.5946536 26.67602308 245.557484   15.7299841    0.7642925           YES
## 6  9.0535479 16.11649095 142.528062    9.1301100    0.5298358           YES
##   Cl_dups_chk Cl_dups_flag SO4_dups_cv SO4_dups_cv_flag
## 1   1.4279429          YES   6.9078502              YES
```

```
## 2   1.1217307          YES   0.9804736             YES
## 3   1.0604602          YES   0.1394063             YES
## 4   1.9303331          YES   1.3391676             YES
## 5   0.8640534          YES   0.5404364             YES
## 6   0.8420997          YES   0.3746505             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 NO, rerun         2 NO, rerun          1    12  16.66667   8.333333
## 2       YES        10       YES         11    12  83.33333  91.666667
```

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_202307_UP_LysA_20cm  8.3626   1.2711 0.2608422  0.03585614
## 27      10_GCW_202307_TR_LysC_20cm  6.5457  62.1675 0.2041703  1.75366714
## 28  10_GCW_202307_TR_LysC_20cm_dup  7.2180  63.0616 0.2251404  1.77888858
## 29     100_SWH_202307_TR_LysB_10cm 21.1419 621.7679 0.6594479 17.53929196
## 30 100_SWH_202307_TR_LysB_10cm_dup 21.4371 628.7818 0.6686556 17.73714528
## 31     101_SWH_202307_TR_LysB_20cm  0.0000 640.1021 0.0000000 18.05647673
##       salinity
## 26 0.002322878
## 27 0.112362672
## 28 0.113978311
## 29 1.123560595
## 30 1.136234713
## 31 1.156690495
```

```
#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_SWH_202307_TR_LysB_20cm_spk   78.6049  630.9976  2.451806  17.79965
## 2      11_GCW_202307_TR_LysC_45cm_spk   84.7580    0.0000  2.643731   0.00000
## 3     111_SWH_202307_WC_LysB_45cm_spk   78.6545  591.9080  2.453353  16.69698
## 4 121_MSM_202307_TR_RHZ_SF_Tree_4_spk 1291.6371 9407.4782 40.288119 265.37315
```

```
## 5  131_MSM_202307_WC_RHZ_Collar_3_spk 1177.2651 9848.3711 36.720683 277.81019
## 6              21_GCW_202307_SW_A_spk  792.8469 4669.5951 24.730097 131.72342
##    salinity
## 1  1.140239
## 2  0.000026
## 3  1.069604
## 4 16.999339
## 5 17.796033
## 6  8.437984
```

```
#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_SWH_202307_TR_LysB_20cm   2.451806
## 2      11_GCW_202307_TR_LysC_45cm   2.643731
## 3     111_SWH_202307_WC_LysB_45cm   2.453353
## 4 121_MSM_202307_TR_RHZ_SF_Tree_4  40.288119
## 5  131_MSM_202307_WC_RHZ_Collar_3  36.720683
## 6              21_GCW_202307_SW_A  24.730097
```

```
#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_SWH_202307_TR_LysB_20cm    0.0000  640.1021  0.00000000  18.0564767
## 2      11_GCW_202307_TR_LysC_45cm   20.0613    4.7001  0.62574236   0.1325839
## 3     111_SWH_202307_WC_LysB_45cm    1.4819  601.9171  0.04622271  16.9793258
## 4 121_MSM_202307_TR_RHZ_SF_Tree_4 1230.7105 9574.1069 38.38772614 270.0735374
## 5  131_MSM_202307_WC_RHZ_Collar_3  951.2624 9260.8436 29.67131628 261.2367729
## 6              21_GCW_202307_SW_A  620.4881 4653.4305 19.35396444 131.2674330
##       salinity SO4_mM_spk
## 1  1.156690495   2.451806
## 2  0.008519081   2.643731
## 3  1.087690200   2.453353
## 4 17.300437168  40.288119
## 5 16.734370385  36.720683
## 6  8.408774914  24.730097
```

```
#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_SWH_202307_TR_LysB_20cm    0.0000  640.1021  0.00000000  18.0564767
## 2      11_GCW_202307_TR_LysC_45cm   20.0613    4.7001  0.62574236   0.1325839
## 3     111_SWH_202307_WC_LysB_45cm    1.4819  601.9171  0.04622271  16.9793258
## 4 121_MSM_202307_TR_RHZ_SF_Tree_4 1230.7105 9574.1069 38.38772614 270.0735374
## 5  131_MSM_202307_WC_RHZ_Collar_3  951.2624 9260.8436 29.67131628 261.2367729
## 6              21_GCW_202307_SW_A  620.4881 4653.4305 19.35396444 131.2674330
##       salinity SO4_mM_spk SO4_spk_Conc
## 1  1.156690495   2.451806 7.797879e-05
## 2  0.008519081   2.643731 7.797879e-05
## 3  1.087690200   2.453353 7.797879e-05
## 4 17.300437168  40.288119 7.797879e-05
## 5 16.734370385  36.720683 7.797879e-05
## 6  8.408774914  24.730097 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     101_SWH_202307_TR_LysB_20cm    0.0000  640.1021  0.00000000  18.0564767
## 2      11_GCW_202307_TR_LysC_45cm   20.0613    4.7001  0.62574236   0.1325839
## 3     111_SWH_202307_WC_LysB_45cm    1.4819  601.9171  0.04622271  16.9793258
```

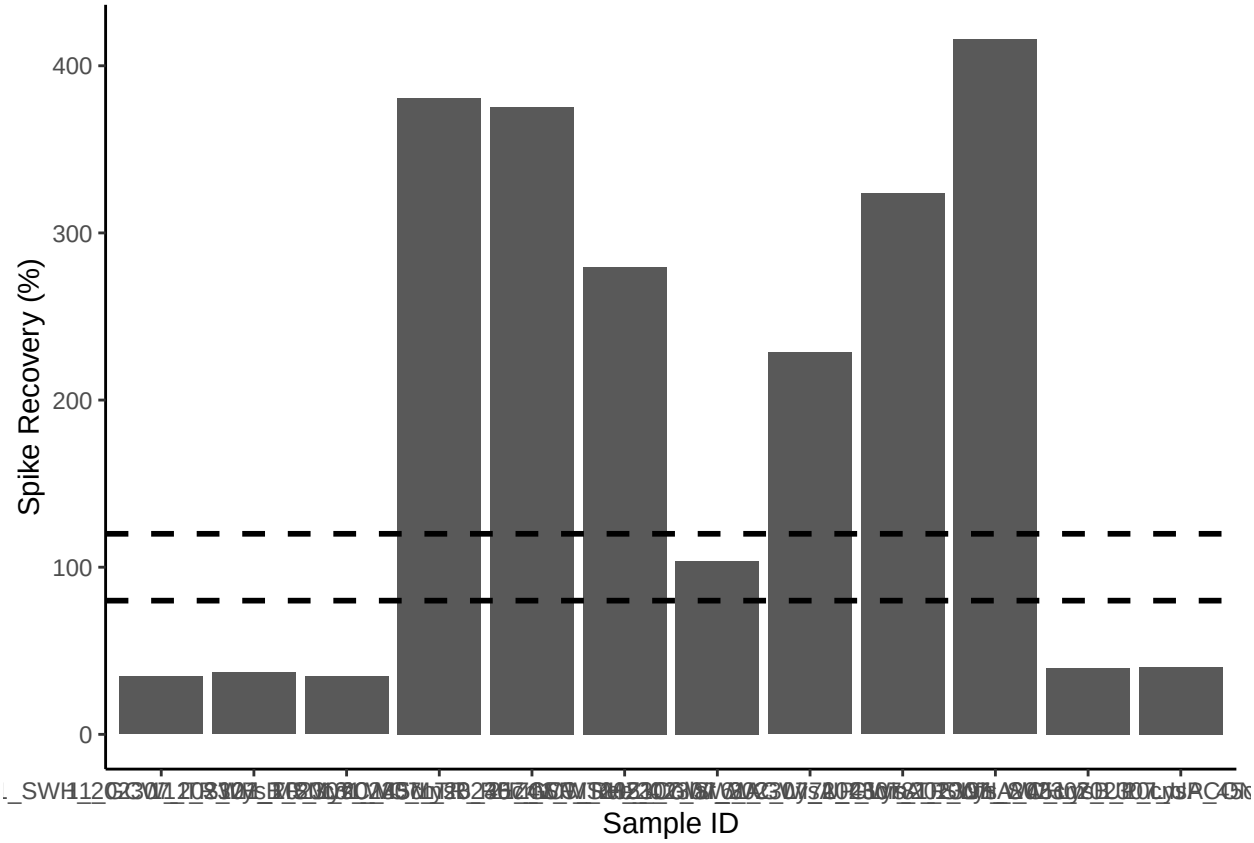
```
## 4 121_MSM_202307_TR_RHZ_SF_Tree_4 1230.7105 9574.1069 38.38772614 270.0735374
## 5  131_MSM_202307_WC_RHZ_Collar_3  951.2624 9260.8436 29.67131628 261.2367729
## 6              21_GCW_202307_SW_A  620.4881 4653.4305 19.35396444 131.2674330
##       salinity SO4_mM_spk SO4_spk_Conc Dilution SampleVol
## 1  1.156690495   2.451806 7.797879e-05        1     1e-06
## 2  0.008519081   2.643731 7.797879e-05        1     1e-06
## 3  1.087690200   2.453353 7.797879e-05        1     1e-06
## 4 17.300437168  40.288119 7.797879e-05        1     1e-06
## 5 16.734370385  36.720683 7.797879e-05        1     1e-06
## 6  8.408774914  24.730097 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     101_SWH_202307_TR_LysB_20cm    0.0000  640.1021  0.00000000  18.0564767
## 2      11_GCW_202307_TR_LysC_45cm   20.0613    4.7001  0.62574236   0.1325839
## 3     111_SWH_202307_WC_LysB_45cm    1.4819  601.9171  0.04622271  16.9793258
## 4 121_MSM_202307_TR_RHZ_SF_Tree_4 1230.7105 9574.1069 38.38772614 270.0735374
## 5  131_MSM_202307_WC_RHZ_Collar_3  951.2624 9260.8436 29.67131628 261.2367729
## 6              21_GCW_202307_SW_A  620.4881 4653.4305 19.35396444 131.2674330
##       salinity SO4_mM_spk SO4_spk_Conc Dilution SampleVol SO4_Total_unspkd
## 1  1.156690495   2.451806 7.797879e-05        1     1e-06     0.000000e+00
## 2  0.008519081   2.643731 7.797879e-05        1     1e-06     6.257424e-07
## 3  1.087690200   2.453353 7.797879e-05        1     1e-06     4.622271e-08
## 4 17.300437168  40.288119 7.797879e-05        1     1e-06     3.838773e-05
## 5 16.734370385  36.720683 7.797879e-05        1     1e-06     2.967132e-05
## 6  8.408774914  24.730097 7.797879e-05        1     1e-06     1.935396e-05
##   SO4_Total_spkd SO4_expctd_spkd spk_recovery SO4_spks_flag
## 1   2.696987e-05    7.797879e-05     34.58616     NO, rerun
## 2   2.908104e-05    7.860453e-05     36.99664     NO, rerun
## 3   2.698688e-05    7.802501e-05     34.58748     NO, rerun
## 4   4.431693e-04    1.163665e-04    380.83920     NO, rerun
## 5   4.039275e-04    1.076501e-04    375.22259     NO, rerun
## 6   2.720311e-04    9.733275e-05    279.48563     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          11    12   91.7
## 2 YES                 1    12    8.33
```

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", "202307", "UP", "LysA", "20cm"), c("10", :
## number of columns of result is not a multiple of vector length (arg 1)
```

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

```
##   Analysis_No Site   Date Zone Replicate Depth  NA  NA
## 1           1  GCW 202307   UP      LysA  20cm   1 GCW
## 2          10  GCW 202307   TR      LysC  20cm  10 GCW
## 3         100  SWH 202307   TR      LysB  10cm 100 SWH
## 4         101  SWH 202307   TR      LysB  20cm 101 SWH
## 5         102  SWH 202307   TR      LysB  45cm 102 SWH
## 6         103  SWH 202307   TR      LysC  10cm 103 SWH
```

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

```
##   Analysis_No Site   Date Zone Replicate Depth  NA  NA
## 1           1  GCW 202307   UP      LysA  20cm   1 GCW
## 2          10  GCW 202307   TR      LysC  20cm  10 GCW
## 3         100  SWH 202307   TR      LysB  10cm 100 SWH
## 4         101  SWH 202307   TR      LysB  20cm 101 SWH
## 5         102  SWH 202307   TR      LysB  45cm 102 SWH
## 6         103  SWH 202307   TR      LysC  10cm 103 SWH
##                     Sample_ID SO4_ppm   Cl_ppm    SO4_mM       Cl_mM
## 1   1_GCW_202307_UP_LysA_20cm  8.3626   1.2711 0.2608422  0.03585614
## 2  10_GCW_202307_TR_LysC_20cm  6.5457  62.1675 0.2041703  1.75366714
## 3 100_SWH_202307_TR_LysB_10cm 21.1419 621.7679 0.6594479 17.53929196
## 4 101_SWH_202307_TR_LysB_20cm  0.0000 640.1021 0.0000000 18.05647673
## 5 102_SWH_202307_TR_LysB_45cm  0.0000 791.2957 0.0000000 22.32145839
## 6 103_SWH_202307_TR_LysC_10cm 46.0312 732.7850 1.4357829 20.67094499
##      salinity
## 1 0.002322878
## 2 0.112362672
## 3 1.123560595
## 4 1.156690495
## 5 1.429897330
## 6 1.324168495
```

Make final dataframe with IDs

```
write.csv(alldat, file="Processed Data/COMPASS_SynopticCB_PW_Processed_Cl_SO4_202307.csv")          #C
```

END