Data Wrangling fpk to TPM

```
library(tidyverse)

## Load dataframes
names(KeggIdsOfInterest)
names(samples.fpk)

## Change Entry to KEGG_ID so col names match
KeggIdsOfInterest <-  KeggIdsOfInterest %>%
  rename('KEGG_ID'= Entry)

## Change colnames samples.fpk, make new df because that kept row1 as sample names
names(samples.fpk) <- samples.fpk[1,]
dfSamples <- samples.fpk[2:6075,]


##################### Convert FPK to TPM #########################################

## Process one column at a time. Function calculates TPM for column.
ColumnTPM <- function(dataframe,i) {
  # Store column fpk counts into a list
  fpkList <- dataframe[1:nrow(dataframe),i]
  # Calculate scaling factor for column/sample
  denom <- sum(as.numeric(dataframe[,i]))/1e6
  # Do maths and convert all fpk values to tpm
  colList <- do.call(c, lapply(fpkList, function(e){
    tpm <- as.numeric(e)/denom
    return(tpm)
  }))

    return(colList)

}

dfTPM <- do.call(data.frame, lapply(2:ncol(dfSamples), ColumnTPM, dataframe=dfSamples))

#################################################################################

# Copy the column names from the original dataframe, add back the KEGG_ID column
colnames(dfTPM) <- colnames(dfSamples[,-1])

dfTPM <- cbind(dfTPM, dfSamples$KEGG_ID)
colnames(dfTPM)[17] <- 'KEGG_ID'
```

```
str(dfTPM)

############ Test Values are correct using random number generator #############
##1-17 for cols: 12, 2, 5, 9, 16
##1-6074 for rows: 5335, 287, 4927, 2869, 1449

dfTPM[,12][5335]
#0
dfTPM[,2][287]
#137.8652
dfTPM[,5][4927]
#693.3421
dfTPM[,9][2869]
#372.0757
dfTPM[,16][1449]
#3.235608

## +1 for col number in dfSamples as df[1]=Kegg IDS
denom1 <- sum(as.numeric(dfSamples[,13]))/1e6
denom2 <- sum(as.numeric(dfSamples[,3]))/1e6
denom3 <- sum(as.numeric(dfSamples[,6]))/1e6
denom4 <- sum(as.numeric(dfSamples[,10]))/1e6
denom5 <- sum(as.numeric(dfSamples[,17]))/1e6

as.numeric(dfSamples[,13][5335])/denom1
#0
as.numeric(dfSamples[,3][287])/denom2
#137.8652
as.numeric(dfSamples[,6][4927])/denom3
#693.3421
as.numeric(dfSamples[,10][2869])/denom4
#372.0757
as.numeric(dfSamples[,17][1449])/denom5
#3.235608

###############################################################################

##Save TPM df as csv
write.csv(dfTPM, file = 'samples_tpm.csv')

# Filter to make dfKEGGofIntFPK with subset of fpk for interesting KEGG ids
keggs <- as.vector(KeggIdsOfInterest$KEGG_ID)

dfKEGGofIntFPK <- dfTPM %>%
```

```
  filter(KEGG_ID %in% keggs)

## Change column name of Kegg ID to match and merge dfs, so have fpk and gene info

dfINTMergedTPM <- merge(dfKEGGofIntFPK,KeggIdsOfInterest,by="KEGG_ID")
write.csv(dfINTMergedTPM, file = 'samples_INT_tpm.csv')

################### Modify dfs for Bubble Plots ###################################
### Sulfur: SOX, aprAB, dsrAB
### Denit: Nap, Nir, Nor, Nos
### Carbon: Aerobic CODH, RuBisCO, mcrABG, Anaerobic CODH/ACS

dfINTMergedTPM <- samples_INT_tpm[,2:22]
str(dfINTMergedTPM)

keggS <- c('K17222','K17223','K17224','K17225','K17226','K17227','K00394','K00395','K11180','K11181')

dfSulfur <- dfINTMergedTPM %>%
  filter(KEGG_ID %in% keggS)

keggD1 <- c('K02567','K00368','K15864','K04748','K04561','K02305','K02448','K02164','K00376','K04747',
          'K00362','K00363','K00374','K02568')

dfDenit1 <- dfINTMergedTPM %>%
  filter(KEGG_ID %in% keggD1)

keggD2 <- c('K02567',
          'K00368',
          'K15864',
          'K04748',
          'K04561',
          'K02305',
          'K02448',
          'K02164',
          'K00376')

dfDenit2 <- dfINTMergedTPM %>%
  filter(KEGG_ID %in% keggD2)

keggC <- c('K01601',
         'K01602',
         'K00399',
         'K00401',
         'K00402',
```

```
        'K00198',
        'K14138',
        'K00194',
        'K00197',
        'K00192',
        'K00193',
        'K03518',
        'K03519',
        'K03520')

dfCarbon <- dfINTMergedTPM %>%
  filter(KEGG_ID %in% keggC)

## dfs to plot: dfSulfur, dfDenit1, dfDenit2, dfCarbon. Need TPM data in one column.

write.csv(dfSulfur, file = 'dfSulfurTPM.csv')
write.csv(dfDenit1, file = 'dfDenit1TPM.csv')
write.csv(dfDenit2, file = 'dfDenit2TPM.csv')
write.csv(dfCarbon, file = 'dfCarbonTPM.csv')
```