macronutrient_update_PER_CELL

Hannah Reich

8/9/2020

PER CELL instead of per cell volume stats & figures for fig s6 (macronutrient) contents

```
# jan 21 2020.
# remaking C/N/P per cell and metal/cell figures (other rmarkdowns were getting super crowded so made a new one)
# set working directory
setwd("~/Desktop/PhD/chapters/chapter 4 - temp treatments/for_coauthors_v1/data_code/code/")
# load packages
library(ggplot2)
library(ggpubr)
```

```
## Loading required package: magrittr
```

```
library(readxl)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(reshape2)
library(cowplot)
```

```
##
## ************************************************************
```

```
## Note: As of version 1.0.0, cowplot does not change the
```

```
##   default ggplot2 theme anymore. To recover the previous
```

```
##   behavior, execute:
##   theme_set(theme_cowplot())
```

```
## ************************************************************
```

```
##
## Attaching package: 'cowplot'
```

```
## The following object is masked from 'package:ggpubr':
##
##     get_legend
```

```
#### part 1, calculate POC/cell & PON/cell
# this part of the script normalizes all of the metals to Particulate Organic Carbon (POC).
# calculation info for the POC and Particulate Organic Nitrogen (PON) is in the excel file "poc_pon"
library(data.table) # for function `fread`
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:reshape2':
##
##     dcast, melt
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
library(broom)      # for function `tidy`
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

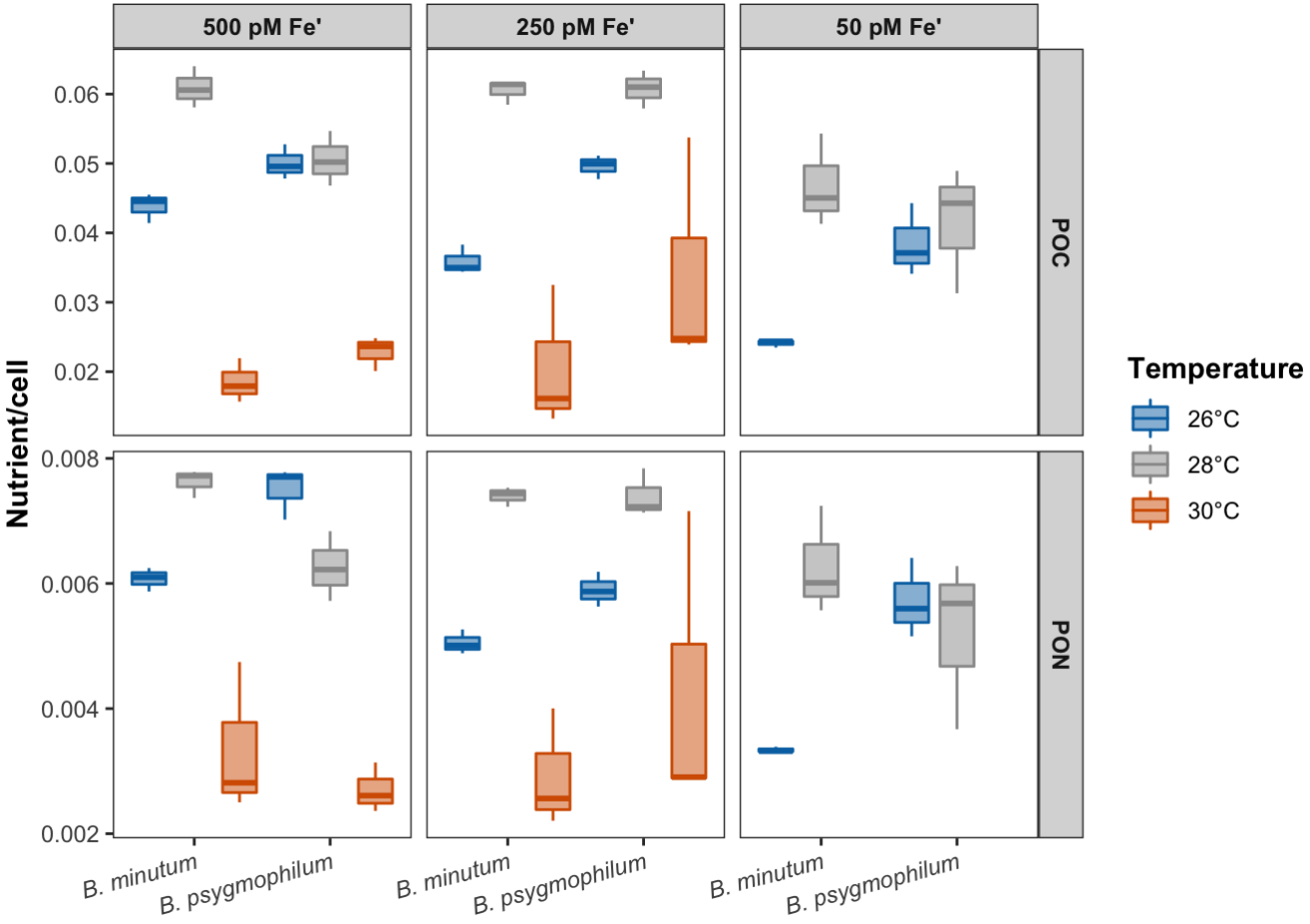
```
library(PMCMR)
```

```
## PMCMR is superseded by PMCMRplus and will be no longer maintained. You may wish to install PMCMRplus instead.
```

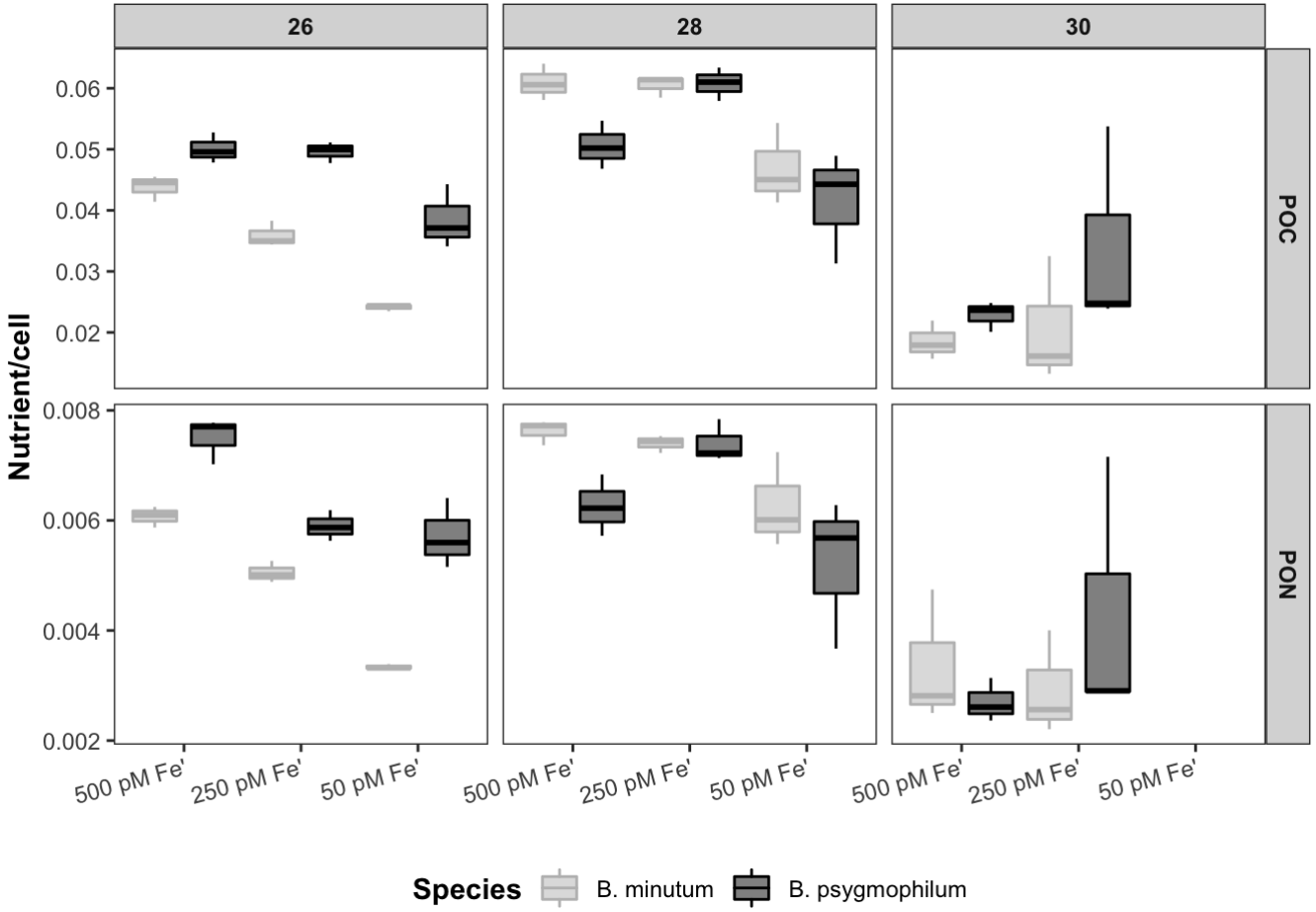
```
# read in the data

dat <- read_excel("~/Desktop/PhD/chapters/chapter 4 - temp treatments/for_coauthors_v1/data_code/poc_pon.xlsx", s
heet = "pocpon_forR")
dat <- as.data.frame(dat)
# normalize the POC and PON content to vol per cell
# load the cell vol data back in
volused <- read_excel("~/Desktop/PhD/chapters/chapter 4 - temp treatments/for_coauthors_v1/data_code/cell_densit
y.xlsx", sheet ="vol")
#all$tmvol <- (all$mLTM*all$TotalVol)
# calculate the number of cells used
volused$celltot_pocpon <- (volused$CellDensity*volused$mLPOCPON)
# merge the data
all <- merge(dat, volused, by = "sampleID")
all <- as.data.frame(all)
# now calculate C and N per uM3 cell
all$C_Cell <- ((all$POC_uM/all$celltot_pocpon)*1000)
all$N_Cell <- ((all$PON_uM/all$celltot_pocpon)*1000)
# write out file with just sample info and N and C per cell
all1 <- all[,c(1:5,19:20)]
#write.csv(all1, "POCPON_cell.csv")
# read in the metals (and phosphorus)
metals <- read_excel("~/Desktop/PhD/chapters/chapter 4 - temp treatments/for_coauthors_v1/data_code/metallome_um
3.xlsx", sheet = "formanova")
# read in the poc and pon per cell
metals <- as.data.frame(metals)
#cn <- read.csv("POCPON_cell.csv", header = TRUE)
cn <- as.data.frame(all1)
# make the dataset rbind'able (i.e., make sure factor levels & colnames will match)
colnames(cn) <- c("sampleID","TreatmentID", "Species", "Ironconc", "Temp", "POC", "PON")
#cn$sample_ID <- seq.int(nrow(cn))
metals$Species <- factor(metals$Species, levels = c("m", "p"))
levels(metals$Species) <- c("min", "psyg")
cn$Ironconc <- factor(cn$Ironconc, levels =c("100", "50", "10"))
levels(cn$Ironconc) <- c("500 pM Fe'", "250 pM Fe'", "50 pM Fe'")
metals$Ironconc <- factor(metals$Ironconc, levels =c("500", "250", "50"))
levels(metals$Ironconc) <- c("500 pM Fe'", "250 pM Fe'", "50 pM Fe'")
# melt them and THEN rbind them (otherwise POC/PON triplicates get duplicated because icpms data has tech reps)
cn_melt <- reshape2::melt(cn, id.vars=c("sampleID","TreatmentID", "Species", "Ironconc", "Temp"), measure.vars=c(
"POC","PON"))
```

```
metal_melt <- reshape2::melt(metals, id.vars = c("Species", "Ironconc", "Temp", "TreatmentID", "sampleID"), measu
re.vars = c("Cobalt", "Copper", "Iron", "Manganese", "Nickel", "Zinc", "Phosphorus"))
# rbind them together
allnuts <- rbind(cn_melt, metal_melt)
allnuts$Temp <- as.factor(allnuts$Temp)
# separate out macronutrients
macro <- dplyr::filter(allnuts, variable %in% c("POC", "PON"))
### make a plot of the macronutrients, this is figure S6
# change species labels
macro$Species <- factor(macro$Species, levels = c("min", "psyg"))
levels(macro$Species) <- c("B. minutum", "B. psygmophilum")
# this is figure s6
mac <- ggplot(data=macro, aes(x=Species, y=value, fill=Temp, color=Temp)) +
  geom_boxplot(alpha = 0.5, aes(colour = factor(Temp)), varwidth = FALSE, position = position_dodge(1, preserve =
"single")) +
  theme_bw() +
  labs(y="Nutrient/cell", x=element_blank()) +
  theme(panel.grid = element_blank(),
        strip.text = element_text(face = "bold"),
        legend.title = element_text(face = "bold"),
        axis.text.x = element_text(angle=15, vjust=1, hjust =1, face = "italic"),
        axis.title = element_text(face = "bold"),
        legend.position = "right") +
  facet_grid(variable~Ironconc, scales = "free_y") +
  scale_color_manual(values = c("#0072B2","#999999","#D55E00"), name = "Temperature", breaks = c("26", "28", "30"
), labels=c("26°C", "28°C", "30°C")) +
  scale_fill_manual(values = c("#0072B2","#999999","#D55E00"), name = "Temperature", breaks = c("26", "28", "30"
), labels=c("26°C", "28°C", "30°C"))
mac
```


500 pM Fe'
250 pM Fe'
50 pM Fe'
POC
PON
Nutrient/cell
0.06
0.05
0.04
0.03
0.02
0.008
0.006
0.004
0.002
B. minutum
B. psygmophilum
Temperature
26°C
28°C
30°C

```
mac2 <- ggplot(data=macro, aes(x=Ironconc, y=value, fill=Species, color=Species)) +
  geom_boxplot(alpha = 0.5, aes(colour = factor(Species)), varwidth = FALSE, position = position_dodge(1, preserv
e = "single")) +
  theme_bw() +
  labs(y="Nutrient/cell", x=element_blank()) +
  theme(panel.grid = element_blank(),
        strip.text = element_text(face = "bold"),
        legend.title = element_text(face = "bold"),
        axis.text.x = element_text(angle=15, vjust=1, hjust =1),
        axis.title = element_text(face = "bold"),
        legend.position = "bottom") +
  facet_grid(variable~Temp, scales = "free_y") +
    scale_color_manual(values = c("grey", "black"), breaks = c("B. minutum", "B. psygmophilum"), name = "Species"
) +
  scale_fill_manual(values = c("grey", "black"), breaks = c("B. minutum", "B. psygmophilum"), name = "Species")
mac2
```




```
# changes made in illustrator: y-axis was revised to include specific units. this wasn't done in R because POC &
 PON had different units from phosphorus.
#save_plot("~/Desktop/PhD/chapters/chapter 4 - temp treatments/for_coauthors_v1/figs/figs_v4_01212020/FigS6_macro
nutrients.pdf", mac2, base_aspect_ratio = 1.6)
```

have the stats in the same place

```
library(PMCMR)
metals$TreatmentID <- as.factor(metals$TreatmentID)
cn$TreatmentID <- as.factor(cn$TreatmentID)
metals$Temp <- as.factor(metals$Temp)
cn$Temp <- as.factor(cn$Temp)
metals$Ironconc <- as.factor(metals$Ironconc)
cn$Ironconc <- as.factor(cn$Ironconc)
# make the sp-specific objects
min_cn <- filter(cn, Species %in% c("min"))
psyg_cn <- filter(cn, Species %in% c("psyg"))

#data is normal but doing non-parametric stats to keep it consistent

### PON
shapiro.test(min_cn$PON)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  min_cn$PON
## W = 0.91983, p-value = 0.0579
```

```
shapiro.test(psyg_cn$PON)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  psyg_cn$PON
## W = 0.88627, p-value = 0.01114
```

```
kruskal.test(min_cn$PON, min_cn$TreatmentID)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  min_cn$PON and min_cn$TreatmentID
## Kruskal-Wallis chi-squared = 21.067, df = 7, p-value = 0.003673
```

```
kruskal.test(min_cn$PON, min_cn$Temp)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  min_cn$PON and min_cn$Temp
## Kruskal-Wallis chi-squared = 16.828, df = 2, p-value = 0.0002218
```

```
kruskal.test(min_cn$PON, min_cn$Ironconc)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  min_cn$PON and min_cn$Ironconc
## Kruskal-Wallis chi-squared = 1.0322, df = 2, p-value = 0.5968
```

```
kruskal.test(psyg_cn$PON, psyg_cn$TreatmentID)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  psyg_cn$PON and psyg_cn$TreatmentID
## Kruskal-Wallis chi-squared = 16.333, df = 7, p-value = 0.02224
```

```
kruskal.test(psyg_cn$PON, psyg_cn$Temp)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  psyg_cn$PON and psyg_cn$Temp
## Kruskal-Wallis chi-squared = 7.2011, df = 2, p-value = 0.02731
```

```
kruskal.test(psyg_cn$PON, psyg_cn$Ironconc)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  psyg_cn$PON and psyg_cn$Ironconc
## Kruskal-Wallis chi-squared = 0.91111, df = 2, p-value = 0.6341
```

```
posthoc.kruskal.dunn.test(PON~TreatmentID, data = min_cn, p.adjust.method = "fdr")
```

```
##
##  Pairwise comparisons using Dunn's-test for multiple
##                          comparisons of independent samples
##
## data:  PON by TreatmentID
##
##            min-10-26 min-10-28 min-100-26 min-100-28 min-100-30 min-50-26
## min-10-28  0.203     -         -          -          -          -
## min-100-26 0.203     1.000     -          -          -          -
## min-100-28 0.033     0.409     0.409      -          -          -
## min-100-30 0.894     0.165     0.165      0.033      -          -
## min-50-26  0.510     0.510     0.510      0.154      0.465      -
## min-50-28  0.061     0.510     0.510      0.817      0.044      0.212
## min-50-30  0.817     0.152     0.152      0.033      0.894      0.409
##            min-50-28
## min-10-28  -
## min-100-26 -
## min-100-28 -
## min-100-30 -
## min-50-26  -
## min-50-28  -
## min-50-30  0.033
##
## P value adjustment method: fdr
```

```
posthoc.kruskal.dunn.test(PON~TreatmentID, data = psyg_cn, p.adjust.method = "fdr")
```

```
##
##  Pairwise comparisons using Dunn's-test for multiple
##                          comparisons of independent samples
##
## data:  PON by TreatmentID
##
##             psyg-10-26 psyg-10-28 psyg-100-26 psyg-100-28 psyg-100-30
## psyg-10-28  1.000      -          -           -           -
## psyg-100-26 0.201      0.201      -           -           -
## psyg-100-28 0.736      0.736      0.394       -           -
## psyg-100-30 0.368      0.368      0.021       0.201       -
## psyg-50-26  0.929      0.929      0.239       0.859       0.311
## psyg-50-28  0.201      0.201      0.989       0.381       0.021
## psyg-50-30  0.929      0.929      0.201       0.601       0.449
##             psyg-50-26 psyg-50-28
## psyg-10-28  -          -
## psyg-100-26 -          -
## psyg-100-28 -          -
## psyg-100-30 -          -
## psyg-50-26  -          -
## psyg-50-28  0.233      -
## psyg-50-30  0.873      0.201
##
## P value adjustment method: fdr
```

```
### POC
shapiro.test(min_cn$POC)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  min_cn$POC
## W = 0.92869, p-value = 0.09114
```

```
shapiro.test(psyg_cn$POC)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  psyg_cn$POC
## W = 0.91343, p-value = 0.04189
```

```
kruskal.test(min_cn$POC, min_cn$TreatmentID)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  min_cn$POC and min_cn$TreatmentID
## Kruskal-Wallis chi-squared = 21.44, df = 7, p-value = 0.003171
```

```
kruskal.test(min_cn$POC, min_cn$Temp)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  min_cn$POC and min_cn$Temp
## Kruskal-Wallis chi-squared = 17.978, df = 2, p-value = 0.0001248
```

```
kruskal.test(min_cn$POC, min_cn$Ironconc)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  min_cn$POC and min_cn$Ironconc
## Kruskal-Wallis chi-squared = 0.32889, df = 2, p-value = 0.8484
```

```
kruskal.test(psyg_cn$POC, psyg_cn$TreatmentID)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  psyg_cn$POC and psyg_cn$TreatmentID
## Kruskal-Wallis chi-squared = 16.64, df = 7, p-value = 0.01987
```

```
kruskal.test(psyg_cn$POC, psyg_cn$Temp)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  psyg_cn$POC and psyg_cn$Temp
## Kruskal-Wallis chi-squared = 8.0456, df = 2, p-value = 0.0179
```

```
kruskal.test(psyg_cn$POC, psyg_cn$Ironconc)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  psyg_cn$POC and psyg_cn$Ironconc
## Kruskal-Wallis chi-squared = 3.5644, df = 2, p-value = 0.1683
```

```
posthoc.kruskal.dunn.test(POC~TreatmentID, data = min_cn, p.adjust.method = "fdr")
```

```
##
##  Pairwise comparisons using Dunn's-test for multiple
##                          comparisons of independent samples
##
## data:  POC by TreatmentID
##
##            min-10-26 min-10-28 min-100-26 min-100-28 min-100-30 min-50-26
## min-10-28  0.287     -         -          -          -          -
## min-100-26 0.298     0.954     -          -          -          -
## min-100-28 0.061     0.457     0.440      -          -          -
## min-100-30 0.658     0.151     0.152      0.023      -          -
## min-50-26  0.595     0.559     0.576      0.171      0.381      -
## min-50-28  0.061     0.440     0.440      0.954      0.023      0.165
## min-50-30  0.721     0.154     0.159      0.023      0.954      0.434
##            min-50-28
## min-10-28  -
## min-100-26 -
## min-100-28 -
## min-100-30 -
## min-50-26  -
## min-50-28  -
## min-50-30  0.023
##
## P value adjustment method: fdr
```

```
posthoc.kruskal.dunn.test(POC~TreatmentID, data = psyg_cn, p.adjust.method = "fdr")
```

```
##
##  Pairwise comparisons using Dunn's-test for multiple
##                          comparisons of independent samples
##
## data:  POC by TreatmentID
##
##             psyg-10-26 psyg-10-28 psyg-100-26 psyg-100-28 psyg-100-30
## psyg-10-28  0.942      -          -           -           -
## psyg-100-26 0.402      0.415      -           -           -
## psyg-100-28 0.402      0.402      0.942       -           -
## psyg-100-30 0.415      0.402      0.113       0.113       -
## psyg-50-26  0.402      0.415      0.954       0.942       0.113
## psyg-50-28  0.113      0.113      0.402       0.402       0.012
## psyg-50-30  0.942      0.942      0.402       0.402       0.402
##             psyg-50-26 psyg-50-28
## psyg-10-28  -          -
## psyg-100-26 -          -
## psyg-100-28 -          -
## psyg-100-30 -          -
## psyg-50-26  -          -
## psyg-50-28  0.402      -
## psyg-50-30  0.402      0.113
##
## P value adjustment method: fdr
```

```
# between species
cn$Species <- as.factor(cn$Species)
kruskal.test(cn$POC, cn$Species)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  cn$POC and cn$Species
## Kruskal-Wallis chi-squared = 1.0629, df = 1, p-value = 0.3025
```

```
kruskal.test(cn$PON, cn$Species)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  cn$PON and cn$Species
## Kruskal-Wallis chi-squared = 0.43537, df = 1, p-value = 0.5094
```

```
summary(min_cn)
```

```
##    sampleID            TreatmentID   Species               Ironconc Temp
##  Length:24          min-10-26 :3    Length:24          500 pM Fe':9   26:9
##  Class :character   min-10-28 :3    Class :character   250 pM Fe':9   28:9
##  Mode  :character   min-100-26:3    Mode  :character   50 pM Fe' :6   30:6
##                     min-100-28:3
##                     min-100-30:3
##                     min-50-26 :3
##                     (Other)   :6
##       POC                PON
##  Min.   :0.01324   Min.   :0.002207
##  1st Qu.:0.02415   1st Qu.:0.003373
##  Median :0.03980   Median :0.005418
##  Mean   :0.03891   Mean   :0.005255
##  3rd Qu.:0.05525   3rd Qu.:0.007231
##  Max.   :0.06402   Max.   :0.007781
##
```

```
summary(psyg_cn)
```

```
##    sampleID              TreatmentID   Species               Ironconc Temp
##  Length:24          psyg-10-26 :3    Length:24          500 pM Fe':9   26:9
##  Class :character   psyg-10-28 :3    Class :character   250 pM Fe':9   28:9
##  Mode  :character   psyg-100-26:3    Mode  :character   50 pM Fe' :6   30:6
##                     psyg-100-28:3
##                     psyg-100-30:3
##                     psyg-50-26 :3
##                     (Other)    :6
##       POC               PON
##  Min.   :0.02010   Min.   :0.002364
##  1st Qu.:0.03341   1st Qu.:0.004785
##  Median :0.04779   Median :0.006029
##  Mean   :0.04350   Mean   :0.005625
##  3rd Qu.:0.05154   3rd Qu.:0.007050
##  Max.   :0.06338   Max.   :0.007840
##
```

history: script updated 4.22.2020 to only include necessary info for ms. original script (~/Desktop/PhD/chapters/chapter 4 - temp treatments/for_coauthors_v1/data_code/code/NEW_fig2_thru_3_nutsPERcell_jan2020) used to be named: nutrients_stats_figs; renamed micronutrients to avoid confusion