3. Worksheet: Basic R

Lauren Albert; Z620: Quantitative Biodiversity, Indiana University

18 January, 2023

OVERVIEW

This worksheet introduces some of the basic features of the R computing environment (http://www.r-project.org). It is designed to be used along side the **3. RStudio** handout in your binder. You will not be able to complete the exercises without the corresponding handout.

Directions:

1. In the Markdown version of this document in your cloned repo, change "Student Name" on line 3 (above) with your name.
2. Complete as much of the worksheet as possible during class.
3. Use the handout as a guide; it contains a more complete description of data sets along with examples of proper scripting needed to carry out the exercises.
4. Answer questions in the worksheet. Space for your answers is provided in this document and is indicated by the ">" character. If you need a second paragraph be sure to start the first line with ">". You should notice that the answer is highlighted in green by RStudio (color may vary if you changed the editor theme).
5. Before you leave the classroom today, it is *imperative* that you **push** this file to your GitHub repo, at whatever stage you are. Ths will enable you to pull your work onto your own computer.
6. When you have completed the worksheet, **Knit** the text and code into a single PDF file by pressing the `Knit` button in the RStudio scripting panel. This will save the PDF output in your '3.RStudio' folder.
7. After Knitting, please submit the worksheet by making a **push** to your GitHub repo and then create a **pull request** via GitHub. Your pull request should include this file (**3.RStudio_Worksheet.Rmd**) with all code blocks filled out and questions answered) and the PDF output of `Knitr` (**3.RStudio_Worksheet.pdf**).

The completed exercise is due on **Wednesday, January 18th, 2023 before 12:00 PM (noon)**.

1) HOW WE WILL BE USING R AND OTHER TOOLS

You are working in an RMarkdown (.Rmd) file. This allows you to integrate text and R code into a single document. There are two major features to this document: 1) Markdown formatted text and 2) "chunks" of R code. Anything in an R code chunk will be interpreted by R when you *Knit* the document.

When you are done, you will *knit* your document together. However, if there are errors in the R code contained in your Markdown document, you will not be able to knit a PDF file. If this happens, you will need to review your code, locate the source of the error(s), and make the appropriate changes. Even if you are able to knit without issue, you should review the knitted document for correctness and completeness before you submit the Worksheet. Next to the `Knit` button in the RStudio scripting panel there is a spell checker button (`ABC`) button.

2) SETTING YOUR WORKING DIRECTORY

In the R code chunk below, please provide the code to: 1) clear your R environment, 2) print your current working directory, and 3) set your working directory to your '3.RStudio' folder.

```
rm(list = ls())
getwd()
```

```
## [1] "/Users/laurenalbert/GitHub/QB2023_Albert/2.Worksheets/3.RStudio"
```

```
setwd("~/GitHub/QB2023_Albert/2.Worksheets/3.RStudio")
```

3) USING R AS A CALCULATOR

To follow up on the pre-class exercises, please calculate the following in the R code chunk below. Feel free to reference the **1. Introduction to version control and computing tools** handout.

1) the volume of a cube with length, l, = 5 (volume = l^3)
2) the area of a circle with radius, r, = 2 (area = pi * r^2).
3) the length of the opposite side of a right-triangle given that the angle, theta, = pi/4. (radians, a.k.a. 45°) and with hypotenuse length sqrt(2) (remember: sin(theta) = opposite/hypotenuse).
4) the log (base e) of your favorite number.

```
5^3
```

```
## [1] 125
```

```
pi * 2^2
```

```
## [1] 12.56637
```

```
sin(pi/4)*sqrt(2)
```

```
## [1] 1
```

```
log(64)
```

```
## [1] 4.158883
```

4) WORKING WITH VECTORS

To follow up on the pre-class exercises, please perform the requested operations in the R-code chunks below.

Basic Features Of Vectors

In the R-code chunk below, do the following: 1) Create a vector `x` consisting of any five numbers. 2) Create a new vector `w` by multiplying `x` by 14 (i.e., "scalar"). 3) Add `x` and `w` and divide by 15.

```
x <- c(8, 24, 56, 79)
w <- x * 14
x + w / 15
```

```
## [1]  15.46667  46.40000 108.26667 152.73333
```

Now, do the following: 1) Create another vector (`k`) that is the same length as `w`. 2) Multiply `k` by `x`. 3) Use the combine function to create one more vector, `d` that consists of any three elements from `w` and any four elements of `k`.

```
k <- c(50, 62, 35, 67)
k * x
```

```
## [1]  400 1488 1960 5293
```

```
d <- c(w[2:4], k[1:4])
```

Summary Statistics of Vectors

In the R-code chunk below, calculate the **summary statistics** (i.e., maximum, minimum, sum, mean, median, variance, standard deviation, and standard error of the mean) for the vector (`v`) provided.

```
v <- c(16.4, 16.0, 10.1, 16.8, 20.5, NA, 20.2, 13.1, 24.8, 20.2, 25.0, 20.5, 30.5, 31.4, 27.1)

max(na.omit(v))
```

```
## [1] 31.4
```

```
min(na.omit(v))
```

```
## [1] 10.1
```

```
sum(na.omit(v))
```

```
## [1] 292.6
```

```
mean(na.omit(v))
```

```
## [1] 20.9
```

```
median(na.omit(v))
```

```
## [1] 20.35
```

```
var(na.omit(v))
```

```
## [1] 39.44
```

```
sd(na.omit(v))
```

```
## [1] 6.280127
```

```
sem <- function(v){
  sd(na.omit(v))/sqrt(length(na.omit(v)))
}
sem(v)
```

```
## [1] 1.678435
```

5) WORKING WITH MATRICES

In the R-code chunk below, do the following: Using a mixture of Approach 1 and 2 from the **3. RStudio** handout, create a matrix with two columns and five rows. Both columns should consist of random numbers. Make the mean of the first column equal to 8 with a standard deviation of 2 and the mean of the second column equal to 25 with a standard deviation of 10.

```
j <- c(rnorm(5, mean = 8, sd = 2))
z <- c(rnorm(5, mean = 25, sd = 10))
t <- cbind(z, j)
```

Question 1: What does the `rnorm` function do? What do the arguments in this function specify? Remember to use `help()` or type `?rnorm`.

> Answer 1: The rnorm function generates a vector of random numbers with a normal distrbutions. Within the function 'n' defines the >the number of random numbers or sample size, 'mean' is the mean value of the random numbers, and 'sd' is the standard deviation of >the random numbers.

In the R code chunk below, do the following: 1) Load `matrix.txt` from the **3.RStudio** data folder as matrix `m`. 2) Transpose this matrix. 3) Determine the dimensions of the transposed matrix.

```
m <- as.matrix(read.table("data/matrix.txt", sep = "\t", header = FALSE))
n <- t(m)
dim(m)
```

```
## [1] 10  5
```

```
dim(n)
```

```
## [1]  5 10
```

Question 2: What are the dimensions of the matrix you just transposed?

> Answer 2: The transposed matrix is 10 columns and 5 rows, while the original imported matrix was 5 columns and 10 rows.

###Indexing a Matrix

In the R code chunk below, do the following: 1) Index matrix `m` by selecting all but the third column. 2) Remove the last row of matrix `m`.

```
n <- m[1:9, c(1:2, 4:5)]
```

6) BASIC DATA VISUALIZATION AND STATISTICAL ANALYSIS

Load Zooplankton Data Set

In the R code chunk below, do the following: 1) Load the zooplankton data set from the **3.RStudio** data folder. 2) Display the structure of this data set.

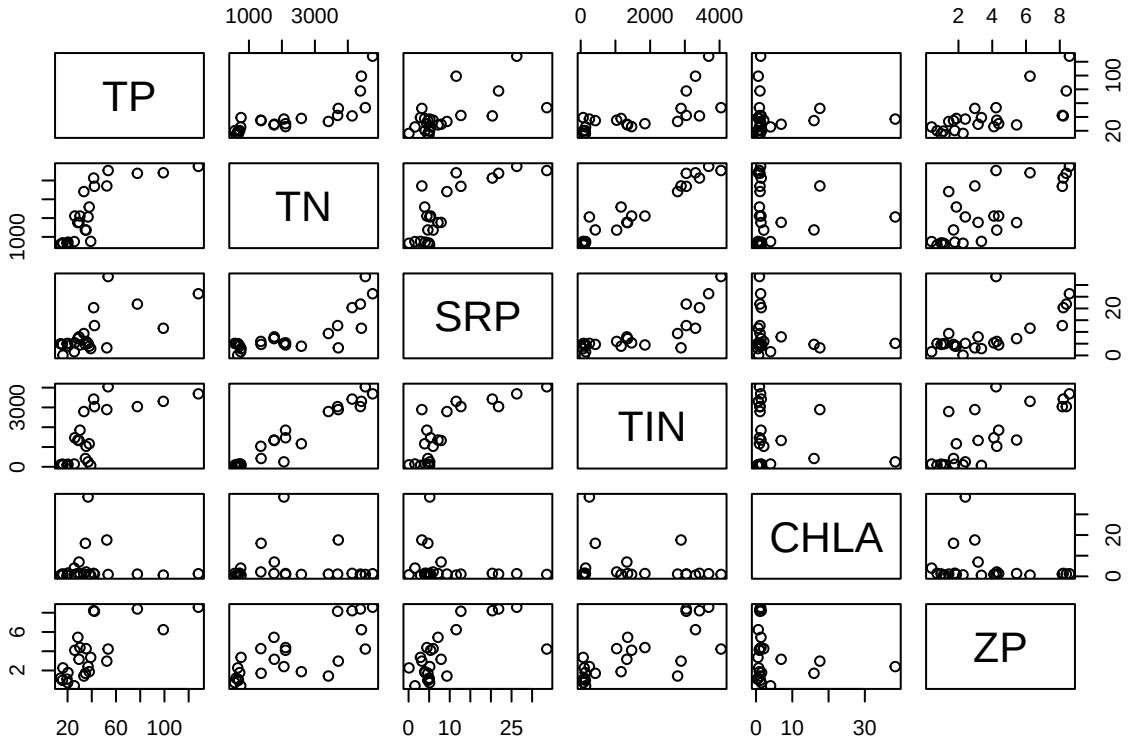
```
meso <- read.table("data/zoop_nuts.txt", sep = "\t", header = TRUE)
str(meso)
```

```
## 'data.frame':    24 obs. of  8 variables:
##  $ TANK: int  34 14 23 16 21 5 25 27 30 28 ...
##  $ NUTS: chr  "L" "L" "L" "L" ...
##  $ TP  : num  20.3 25.6 14.2 39.1 20.1 ...
##  $ TN  : num  720 750 610 761 570 ...
##  $ SRP : num  4.02 1.56 4.97 2.89 5.11 4.68 5 0.1 7.9 3.92 ...
##  $ TIN : num  131.6 141.1 107.7 71.3 80.4 ...
##  $ CHLA: num  1.52 4 0.61 0.53 1.44 1.19 0.37 0.72 6.93 0.94 ...
##  $ ZP  : num  1.781 0.409 1.201 3.36 0.733 ...
```

Correlation

In the R-code chunk below, do the following: 1) Create a matrix with the numerical data in the `meso` dataframe. 2) Visualize the pairwise **bi-plots** of the six numerical variables. 3) Conduct a simple **Pearson's correlation** analysis.

```
meso.num <- meso[, 3:8]
pairs(meso.num)
```




```
cor1 <- cor(meso.num)
print(cor1, digits = 3)
```

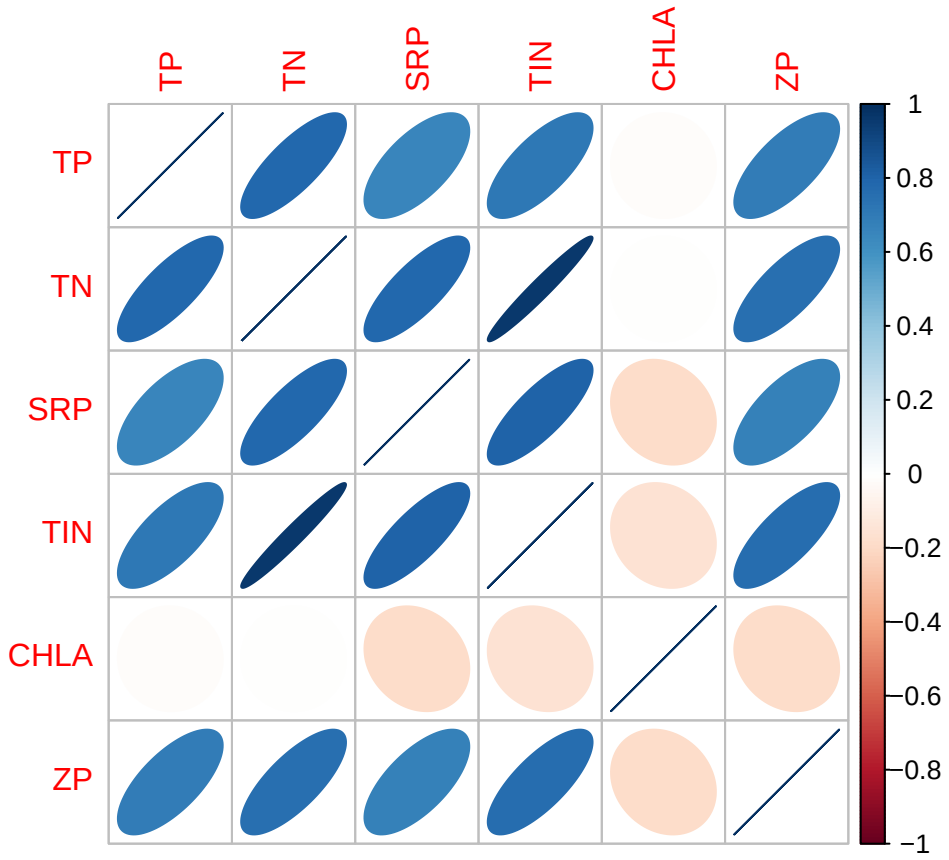
```
##            TP       TN    SRP    TIN      CHLA     ZP
## TP     1.0000  0.78651  0.654  0.717 -0.01666  0.697
## TN     0.7865  1.00000  0.784  0.969 -0.00447  0.756
## SRP    0.6541  0.78419  1.000  0.801 -0.18915  0.676
## TIN    0.7171  0.96900  0.801  1.000 -0.15688  0.761
## CHLA -0.0167 -0.00447 -0.189 -0.157  1.00000 -0.183
## ZP     0.6975  0.75625  0.676  0.761 -0.18260  1.000
```

```
require(corrplot)
```

```
## Loading required package: corrplot
```

```
## corrplot 0.92 loaded
```

```
corrplot(cor1, method = "ellipse")
```



Question 3: Describe some of the general features based on the visualization and correlation analysis above?

> Answer 3: Generally, there appears to be several positive correlations between the variables in the data set. In the visualization with ellipses, the direction and strength of the correlation is demonstrated, especially between total nitrogen >concentration and in zooplankton biomass.

In the R code chunk below, do the following: 1) Redo the correlation analysis using the `corr.test()` function in the `psych` package with the following options: method = "pearson", adjust = "BH". 2) Now, redo this correlation analysis using a non-parametric method. 3) Use the print command from the handout to see the results of each correlation analysis.

```
require("psych")
```

```
## Loading required package: psych
```

```
cor2 <- corr.test(meso.num, method = "pearson", adjust = "BH")
print(cor2, digits = 3)
```

```
## Call:corr.test(x = meso.num, method = "pearson", adjust = "BH")
## Correlation matrix
##          TP     TN    SRP    TIN   CHLA     ZP
## TP    1.000  0.787  0.654  0.717 -0.017  0.697
## TN    0.787  1.000  0.784  0.969 -0.004  0.756
```

```
## SRP   0.654  0.784  1.000  0.801 -0.189  0.676
## TIN   0.717  0.969  0.801  1.000 -0.157  0.761
## CHLA -0.017 -0.004 -0.189 -0.157  1.000 -0.183
## ZP    0.697  0.756  0.676  0.761 -0.183  1.000
## Sample Size
## [1] 24
## Probability values (Entries above the diagonal are adjusted for multiple tests.)
##         TP    TN   SRP   TIN  CHLA    ZP
## TP   0.000 0.000 0.001 0.000 0.983 0.000
## TN   0.000 0.000 0.000 0.000 0.983 0.000
## SRP  0.001 0.000 0.000 0.000 0.491 0.000
## TIN  0.000 0.000 0.000 0.000 0.536 0.000
## CHLA 0.938 0.983 0.376 0.464 0.000 0.491
## ZP   0.000 0.000 0.000 0.000 0.393 0.000
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

```
cor3 <- corr.test(meso.num, method = "kendall", adjust = "BH")
print(cor3, digits = 3)
```

```
## Call:corr.test(x = meso.num, method = "kendall", adjust = "BH")
## Correlation matrix
##         TP    TN    SRP   TIN   CHLA     ZP
## TP   1.000 0.739  0.391 0.577  0.044  0.536
## TN   0.739 1.000  0.478 0.809  0.015  0.551
## SRP  0.391 0.478  1.000 0.563 -0.066  0.449
## TIN  0.577 0.809  0.563 1.000  0.044  0.548
## CHLA 0.044 0.015 -0.066 0.044  1.000 -0.051
## ZP   0.536 0.551  0.449 0.548 -0.051  1.000
## Sample Size
## [1] 24
## Probability values (Entries above the diagonal are adjusted for multiple tests.)
##         TP    TN   SRP   TIN  CHLA    ZP
## TP   0.000 0.000 0.088 0.014 0.899 0.015
## TN   0.000 0.000 0.034 0.000 0.946 0.014
## SRP  0.059 0.018 0.000 0.014 0.899 0.046
## TIN  0.003 0.000 0.004 0.000 0.899 0.014
## CHLA 0.839 0.946 0.760 0.839 0.000 0.899
## ZP   0.007 0.005 0.028 0.006 0.813 0.000
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

Question 4: Describe what you learned from `corr.test`. Specifically, are the results sensitive to whether you use parametric (i.e., Pearson's) or non-parametric methods? When should one use non-parametric methods instead of parametric methods? With the Pearson's method, is there evidence for false discovery rate due to multiple comparisons? Why is false discovery rate important?

> Answer 4: When using corr.test, it is important to choose the correct method based on the data being assessed. In the example >above, the results changed between the Pearson (parametric) and kendall (non-parametric) methods. Non-parametric methods are >appropriate for use when looking at rank-based correlations, meaning the variables or groups are ranked. In using parametric >methods, the variables or groups are discrete. With the Pearson's method the Benjamin-Hochberg corrected P values adjusts for the >false discovery rate. The false discovery

rate is important because it can distinguish between a truly significant p-value from >one that arose by chance because of an increasing number of tests performed.

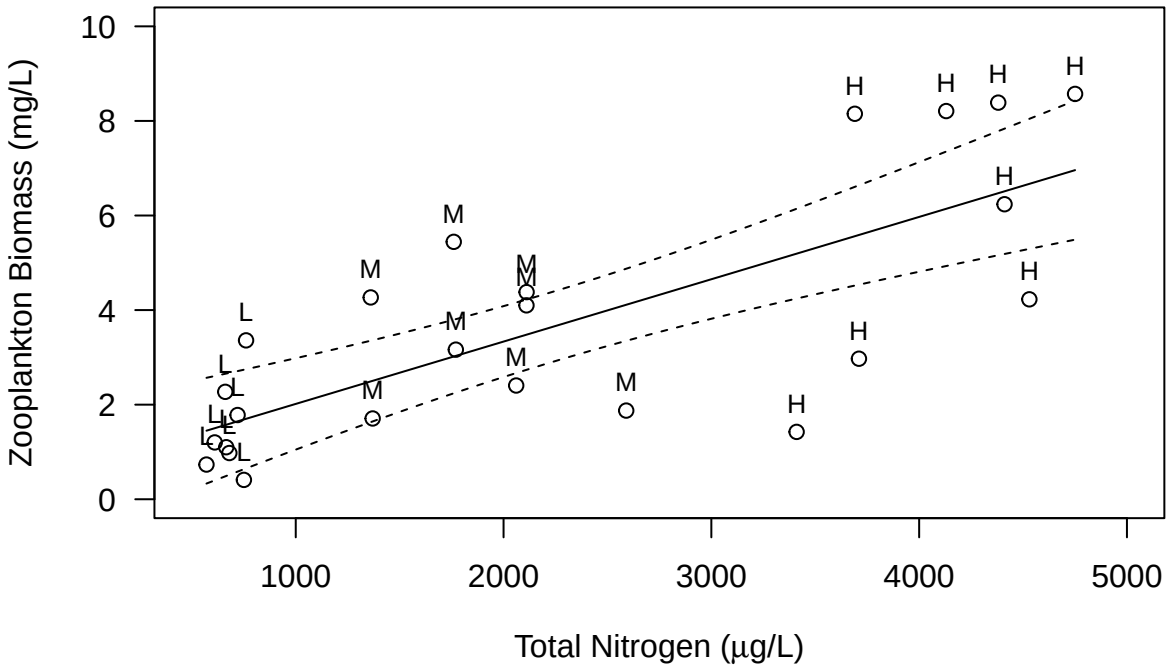
Linear Regression

In the R code chunk below, do the following: 1) Conduct a linear regression analysis to test the relationship between total nitrogen (TN) and zooplankton biomass (ZP). 2) Examine the output of the regression analysis. 3) Produce a plot of this regression analysis including the following: categorically labeled points, the predicted regression line with 95% confidence intervals, and the appropriate axis labels.

```
require("corrplot")
fitreg <-lm(ZP~TN, data = meso)
summary(fitreg)
```

```
##
## Call:
## lm(formula = ZP ~ TN, data = meso)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7690 -0.8491 -0.0709  1.6238  2.5888
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.6977712  0.6496312   1.074    0.294
## TN          0.0013181  0.0002431   5.421 1.91e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.75 on 22 degrees of freedom
## Multiple R-squared:  0.5719, Adjusted R-squared:  0.5525
## F-statistic: 29.39 on 1 and 22 DF,  p-value: 1.911e-05
```

```
plot(meso$TN, meso$ZP, ylim = c(0,10), xlim = c(500, 5000),
      xlab = expression(paste("Total Nitrogen (", mu, "g/L)")),
      ylab = "Zooplankton Biomass (mg/L)", las =1)

text(meso$TN, meso$ZP, meso$NUTS, pos = 3, cex = 0.8)

newTN <- seq(min(meso$TN), max(meso$TN), 10)

regline <- predict(fitreg, newdata = data.frame(TN = newTN))
lines(newTN, regline)

conf95 <- predict(fitreg, newdata = data.frame(TN = newTN), interval = c("confidence"), level = 0.95, ty
matlines(newTN, conf95[, c("lwr","upr")], type ="l", lty = 2, lwd = 1, col = "black")
```




```
##have to run entire chunk of code together, not as separate lines#
```

Question 5: Interpret the results from the regression model

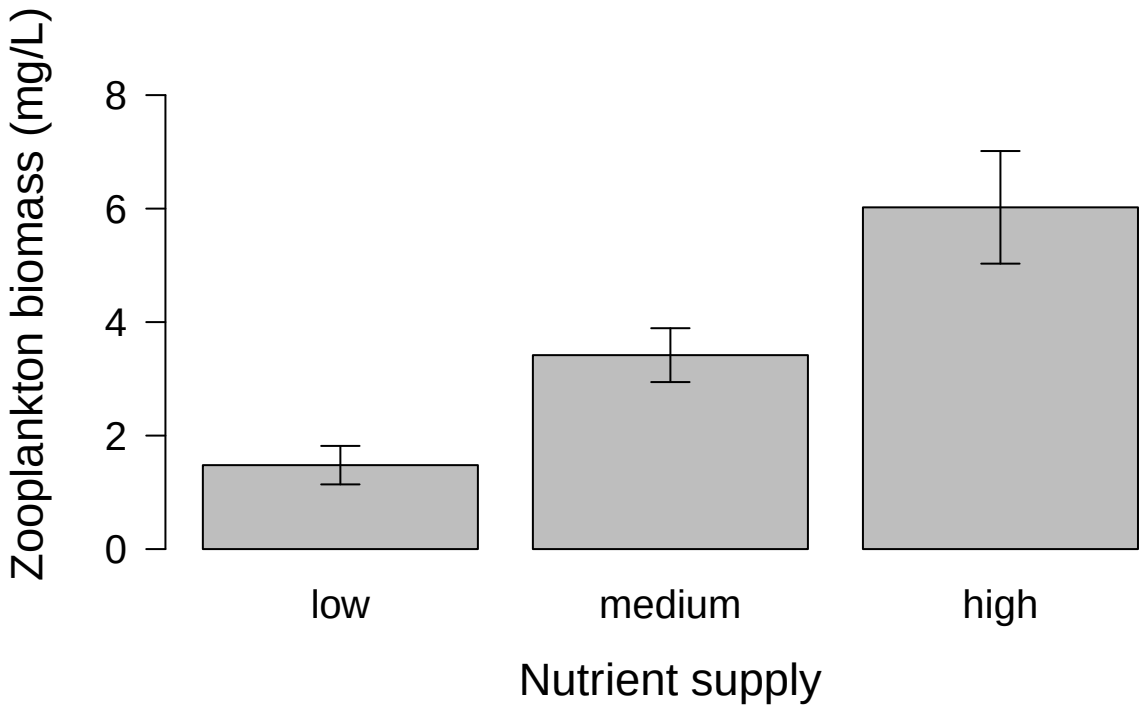
> Answer 5: The linear regression model demonstrates a significant correlation between total nitrogen and total zooplankton >biomass. At high total nitrogen concentrations there is greater zooplankton biomass, whereas mesocosms with lower total nitrogen >concentrations had less zooplankton biomass.

Analysis of Variance (ANOVA)

Using the R code chunk below, do the following: 1) Order the nutrient treatments from low to high (see handout). 2) Produce a barplot to visualize zooplankton biomass in each nutrient treatment. 3) Include error bars (+/- 1 sem) on your plot and label the axes appropriately. 4) Use a one-way analysis of variance (ANOVA) to test the null hypothesis that zooplankton biomass is affected by the nutrient treatment.

```
NUTS <- factor(meso$NUTS, levels = c('L', 'M', 'H'))

zp.means <- tapply(meso$ZP, NUTS, mean)
sem <- function(x){
  sd(na.omit(x))/sqrt(length(na.omit(x)))
}
zp.sem <- tapply(meso$ZP, NUTS, sem)
```

```
bp <- barplot(zp.means, ylim = c(0, round(max(meso$ZP), digits =0)),
                  pch = 15, cex = 1.25, las = 1, cex.lab = 1.4, cex.axis = 1.25,
                  xlab = "Nutrient supply",
                  ylab = "Zooplankton biomass (mg/L)",
                  names.arg = c("low", "medium", "high"))

arrows(x0 = bp, y0 = zp.means, y1 = zp.means - zp.sem, angle = 90, length = 0.1, lwd = 1)
arrows(x0 = bp, y0 = zp.means, y1 = zp.means + zp.sem, angle = 90, length = 0.1, lwd =1)
```




```
fitanova <- aov(ZP ~ NUTS, data = meso)
summary(fitanova)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## NUTS         2  83.15   41.58   11.77 0.000372 ***
## Residuals   21  74.16    3.53
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

SYNTHESIS: SITE-BY-SPECIES MATRIX

In the R code chunk below, load the zoops.txt data set in your **3.RStudio** data folder. Create a site-by-species matrix (or dataframe) that does *not* include TANK or NUTS. The remaining columns of data refer to the biomass (µg/L) of different zooplankton taxa:

- CAL = calanoid copepods
- DIAP = *Diaphanasoma* sp.
- CYL = cyclopoid copepods
- BOSM = *Bosmina* sp.
- SIMO = *Simocephallus* sp.
- CERI = *Ceriodaphnia* sp.
- NAUP = naupuli (immature copepod)
- DLUM = *Daphnia lumholtzi*
- CHYD = *Chydorus* sp.] ***Question 6***: With the visualization and statistical tools that we learned about in the **3. RStudio** handout, use the site-by-species matrix to assess whether and how different zooplankton taxa were responsible for the total biomass (ZP) response to nutrient enrichment. Describe what you learned below in the "Answer" section and include appropriate code in the R chunk.

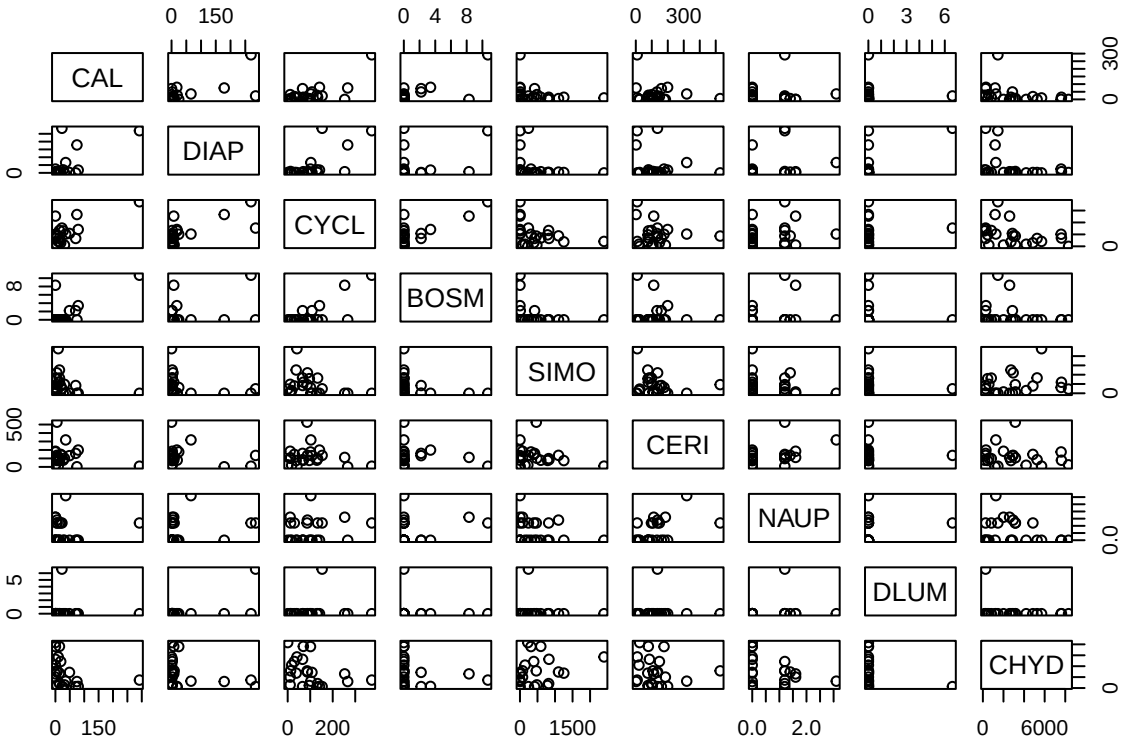
```
zoop <- read.table("data/zoops.txt", sep = "\t", header = TRUE)
str(zoop)
```

```
## 'data.frame':    24 obs. of  11 variables:
##  $ TANK: int  5 14 16 21 23 25 27 34 12 15 ...
##  $ NUTS: chr  "L" "L" "L" "L" ...
##  $ CAL : num  70.5 27.1 5.3 79.2 31.4 22.7 0 35.7 74.8 5.3 ...
##  $ DIAP: num  0 19.2 8.8 17.9 0 ...
##  $ CYCL: num  66.1 129.6 12.7 141.3 11 ...
##  $ BOSM: num  2.2 0 0 3.4 0 0 0 0 0 0 ...
##  $ SIMO: num  417.8 0 73.1 0 482 ...
##  $ CERI: num  159.8 79.4 107.5 199 101.9 ...
##  $ NAUP: num  0 0 1.2 0 0 1.2 1.6 3.1 0 1.4 ...
##  $ DLUM: num  0 0 0 0 0 6.6 0 0 0 0 ...
##  $ CHYD: num  267 159 3158 298 580 ...
```

```
zooptaxa <- zoop[,3:11]
str(zooptaxa)
```

```
## 'data.frame':    24 obs. of  9 variables:
##  $ CAL : num  70.5 27.1 5.3 79.2 31.4 22.7 0 35.7 74.8 5.3 ...
##  $ DIAP: num  0 19.2 8.8 17.9 0 ...
##  $ CYCL: num  66.1 129.6 12.7 141.3 11 ...
##  $ BOSM: num  2.2 0 0 3.4 0 0 0 0 0 0 ...
##  $ SIMO: num  417.8 0 73.1 0 482 ...
##  $ CERI: num  159.8 79.4 107.5 199 101.9 ...
##  $ NAUP: num  0 0 1.2 0 0 1.2 1.6 3.1 0 1.4 ...
##  $ DLUM: num  0 0 0 0 0 6.6 0 0 0 0 ...
##  $ CHYD: num  267 159 3158 298 580 ...
```

```
pairs(zooptaxa)
```




```
cor4 <- cor(zooptaxa)
print(cor4, digits = 3)
```

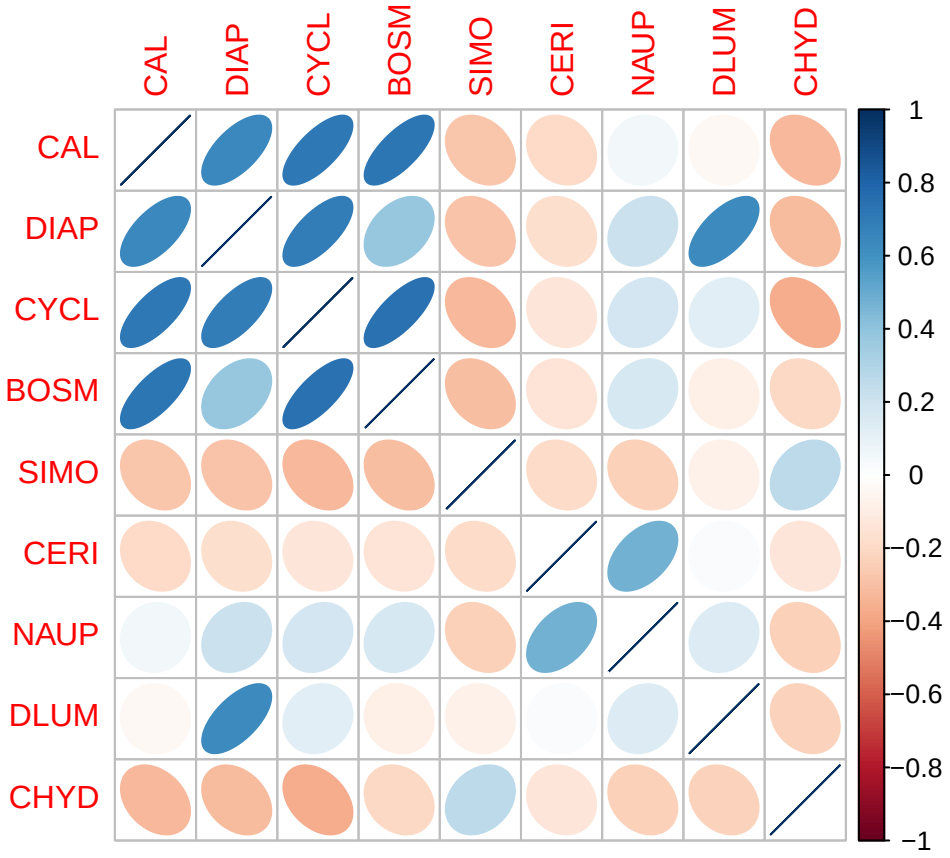
```
##          CAL   DIAP   CYCL    BOSM    SIMO    CERI    NAUP    DLUM   CHYD
## CAL   1.0000  0.643  0.712  0.7281 -0.2715 -0.1912  0.0577 -0.0335 -0.322
## DIAP  0.6426  1.000  0.694  0.3811 -0.2865 -0.1723  0.2168  0.6367 -0.314
## CYCL  0.7119  0.694  1.000  0.7467 -0.3248 -0.1322  0.1856  0.1252 -0.369
## BOSM  0.7281  0.381  0.747  1.0000 -0.3083 -0.1414  0.1789 -0.0863 -0.206
## SIMO -0.2715 -0.287 -0.325 -0.3083  1.0000 -0.1825 -0.2368 -0.0766  0.262
## CERI -0.1912 -0.172 -0.132 -0.1414 -0.1825  1.0000  0.4745  0.0202 -0.135
## NAUP  0.0577  0.217  0.186  0.1789 -0.2368  0.4745  1.0000  0.1475 -0.238
## DLUM -0.0335  0.637  0.125 -0.0863 -0.0766  0.0202  0.1475  1.0000 -0.224
## CHYD -0.3217 -0.314 -0.369 -0.2062  0.2624 -0.1354 -0.2377 -0.2240  1.000
```

```
cor5 <- corr.test(zooptaxa, method = "pearson", adjust = "BH")
print(cor5, digits = 3)
```

```
## Call:corr.test(x = zooptaxa, method = "pearson", adjust = "BH")
## Correlation matrix
##         CAL   DIAP   CYCL   BOSM   SIMO   CERI   NAUP   DLUM   CHYD
## CAL   1.000  0.643  0.712  0.728 -0.271 -0.191  0.058 -0.034 -0.322
## DIAP  0.643  1.000  0.694  0.381 -0.287 -0.172  0.217  0.637 -0.314
## CYCL  0.712  0.694  1.000  0.747 -0.325 -0.132  0.186  0.125 -0.369
## BOSM  0.728  0.381  0.747  1.000 -0.308 -0.141  0.179 -0.086 -0.206
```

```
## SIMO -0.271 -0.287 -0.325 -0.308  1.000 -0.183 -0.237 -0.077  0.262
## CERI -0.191 -0.172 -0.132 -0.141 -0.183  1.000  0.475  0.020 -0.135
## NAUP  0.058  0.217  0.186  0.179 -0.237  0.475  1.000  0.148 -0.238
## DLUM -0.034  0.637  0.125 -0.086 -0.077  0.020  0.148  1.000 -0.224
## CHYD -0.322 -0.314 -0.369 -0.206  0.262 -0.135 -0.238 -0.224  1.000
## Sample Size
## [1] 24
## Probability values (Entries above the diagonal are adjusted for multiple tests.)
##        CAL  DIAP  CYCL  BOSM  SIMO  CERI  NAUP  DLUM  CHYD
## CAL  0.000 0.005 0.001 0.001 0.479 0.580 0.835 0.901 0.395
## DIAP 0.001 0.000 0.002 0.298 0.449 0.582 0.556 0.005 0.395
## CYCL 0.000 0.000 0.000 0.001 0.395 0.646 0.580 0.650 0.306
## BOSM 0.000 0.066 0.000 0.000 0.395 0.646 0.580 0.774 0.572
## SIMO 0.199 0.175 0.122 0.143 0.000 0.580 0.531 0.788 0.485
## CERI 0.371 0.421 0.538 0.510 0.393 0.000 0.098 0.925 0.646
## NAUP 0.789 0.309 0.385 0.403 0.265 0.019 0.000 0.646 0.531
## DLUM 0.876 0.001 0.560 0.688 0.722 0.925 0.491 0.000 0.554
## CHYD 0.125 0.136 0.076 0.334 0.216 0.528 0.263 0.293 0.000
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

```
require("corrplot")
corrplot(cor4, method = "ellipse")
```



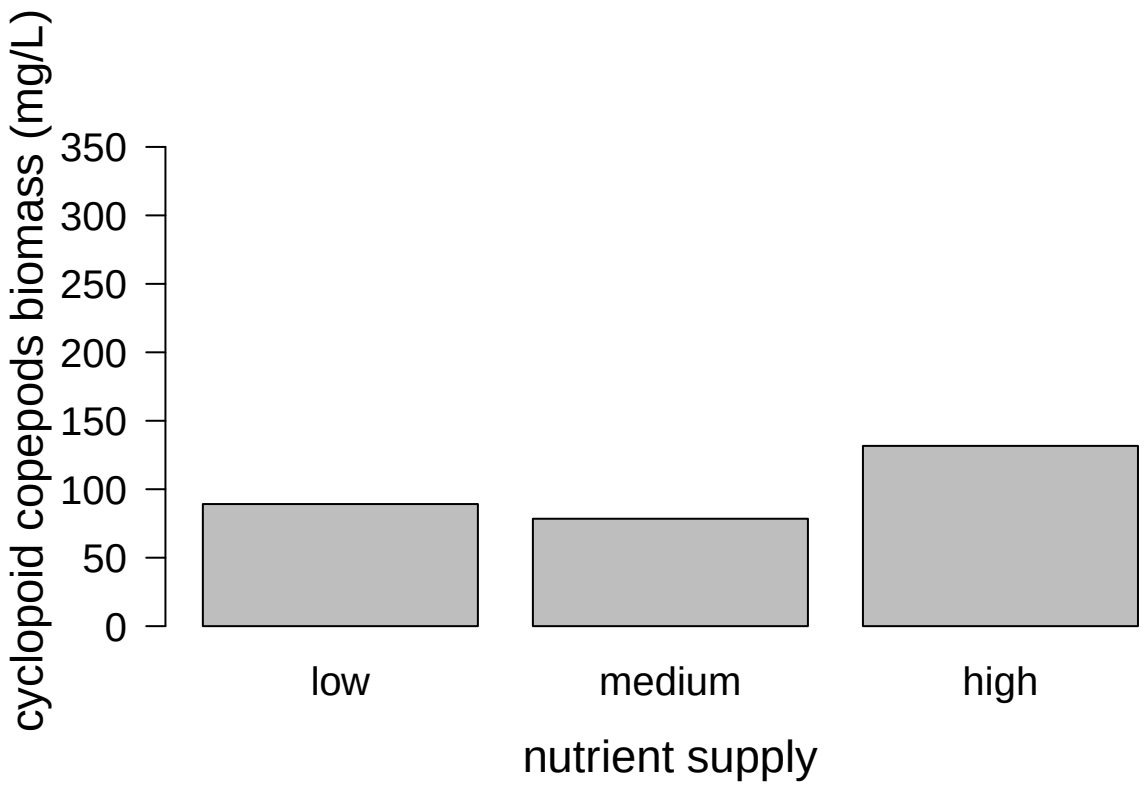
```
nuts <- read.table("data/nuts.txt", sep = "\t", header = TRUE)
str(nuts)
```

```
## 'data.frame':    24 obs. of  7 variables:
##  $ TANK: int  5 14 16 21 23 25 27 34 12 15 ...
##  $ NUTS: chr  "L" "L" "L" "L" ...
##  $ TP  : num  15.8 25.6 39.1 20.1 14.2 ...
##  $ TN  : num  680 750 761 570 610 ...
##  $ SRP : num  4.68 1.56 2.89 5.11 4.97 5 0.1 4.02 4.69 5.96 ...
##  $ TIN : num  135.8 141.1 71.3 80.4 107.7 ...
##  $ CHLA: num  1.19 4 0.53 1.44 0.61 ...
```

```
totalnuts <- nuts[,2]

zooptaxanuts <- cbind(zooptaxa, totalnuts)
str(zooptaxanuts)
```

```
## 'data.frame':    24 obs. of  10 variables:
##  $ CAL      : num  70.5 27.1 5.3 79.2 31.4 22.7 0 35.7 74.8 5.3 ...
##  $ DIAP     : num  0 19.2 8.8 17.9 0 ...
##  $ CYCL     : num  66.1 129.6 12.7 141.3 11 ...
##  $ BOSM     : num  2.2 0 0 3.4 0 0 0 0 0 0 ...
##  $ SIMO     : num  417.8 0 73.1 0 482 ...
##  $ CERI     : num  159.8 79.4 107.5 199 101.9 ...
##  $ NAUP     : num  0 0 1.2 0 0 1.2 1.6 3.1 0 1.4 ...
##  $ DLUM     : num  0 0 0 0 0 6.6 0 0 0 0 ...
##  $ CHYD     : num  267 159 3158 298 580 ...
##  $ totalnuts: chr  "L" "L" "L" "L" ...
```

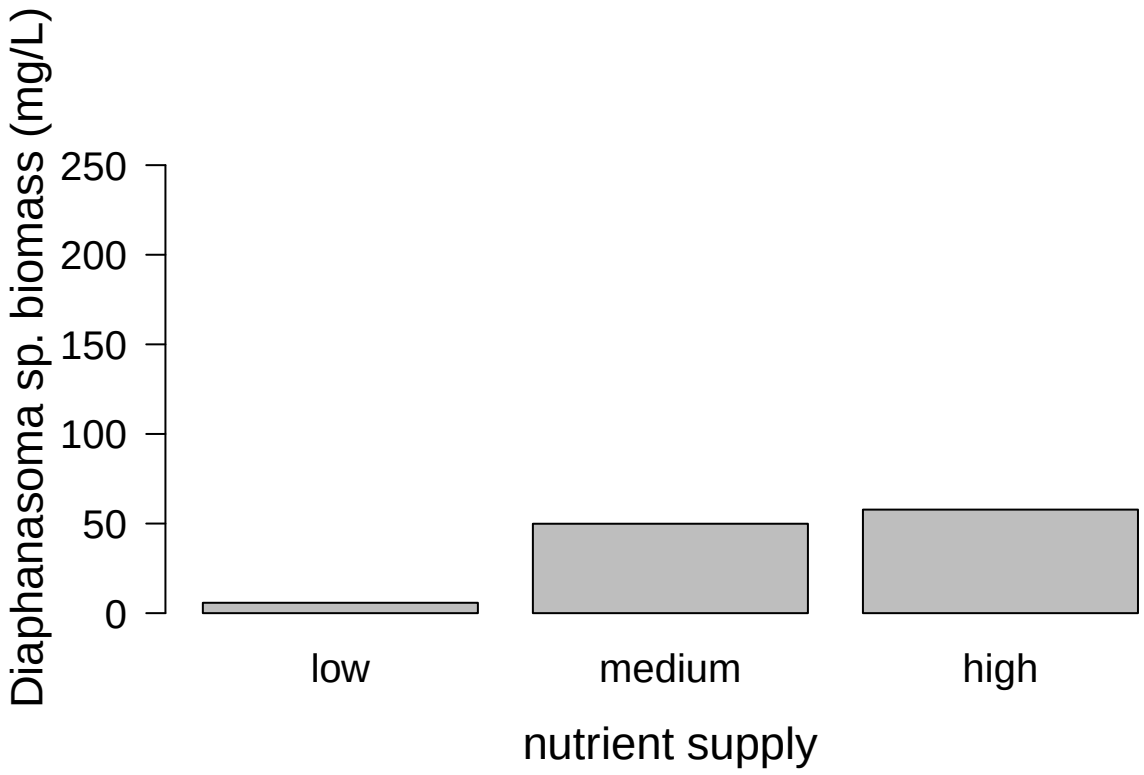
```
nuts <- factor(zooptaxanuts$totalnuts, levels = c('L','M','H'))
#CYCL#
cycl.means <- tapply(zooptaxanuts$CYCL, totalnuts, mean)
sem <- function(x){
  sd(na.omit(x))/sqrt(length(na.omit(x)))
}
cycl.sem <- tapply(zooptaxanuts$CYCL, totalnuts, sem)

bp <- barplot(cycl.means, ylim =c(0, round(max(zooptaxanuts$CYCL), digits = 0)),
              pch = 15, cex = 1.25, las = 1, cex.lab = 1.4, cex.axis = 1.25,
              xlab = "nutrient supply",
              ylab = "cyclopoid copepods biomass (mg/L)",
              names.arg = c("low", "medium", "high"))
```




```
fitanova1 <- aov(CYCL ~ totalnuts, data = zooptaxanuts)
summary(fitanova1)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## totalnuts    2  12669    6335   0.756  0.482
## Residuals   21 175969    8379
```

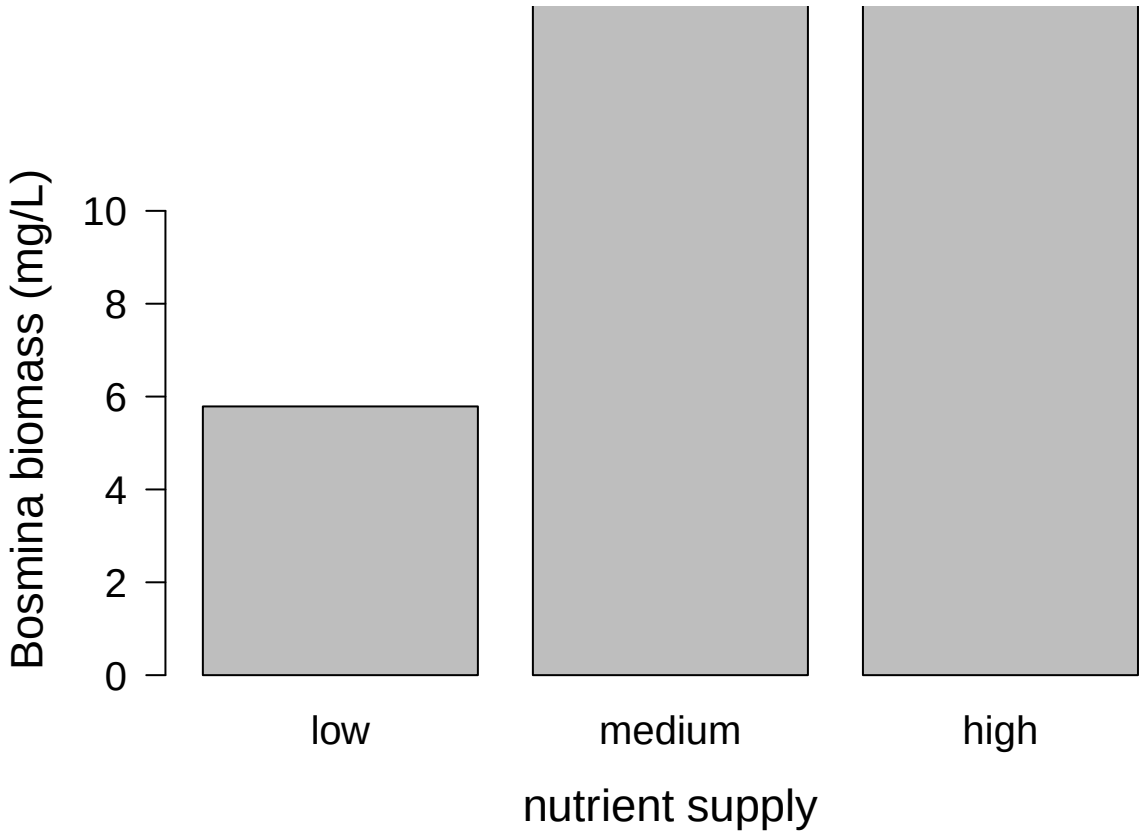
```
#DIAP#
diap.means <- tapply(zooptaxanuts$DIAP, totalnuts, mean)
sem <- function(x){
  sd(na.omit(x))/sqrt(length(na.omit(x)))
}
diap.sem <- tapply(zooptaxanuts$DIAP, totalnuts, sem)
bp <- barplot(diap.means, ylim =c(0, round(max(zooptaxanuts$DIAP), digits = 0)),
              pch = 15, cex = 1.25, las = 1, cex.lab = 1.4, cex.axis = 1.25,
              xlab = "nutrient supply",
              ylab = "Diaphanasoma sp. biomass (mg/L)",
              names.arg = c("low", "medium", "high"))
```




```
fitanova2 <- aov(DIAP ~ totalnuts, data = zooptaxanuts)
summary(fitanova2)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## totalnuts    2  12566    6283   0.911  0.417
## Residuals   21 144826    6896
```

```
#BOSM#
bosm.means <- tapply(zooptaxanuts$BOSM, totalnuts, mean)
sem <- function(x){
  sd(na.omit(x))/sqrt(length(na.omit(x)))
}
bosm.sem <- tapply(zooptaxanuts$BOSM, totalnuts, sem)

bp <- barplot(diap.means, ylim =c(0, round(max(zooptaxanuts$BOSM), digits = 0)),
              pch = 15, cex = 1.25, las = 1, cex.lab = 1.4, cex.axis = 1.25,
              xlab = "nutrient supply",
              ylab = "Bosmina biomass (mg/L)",
              names.arg = c("low", "medium", "high"))
```




```
fitanova3 <- aov(BOSM ~ totalnuts, data = zooptaxanuts)
summary(fitanova3)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## totalnuts    2   3.41   1.703   0.209  0.813
## Residuals   21 171.29   8.157
```

Answer 6: I believe there are key relationships between different zooplankton taxa and nutrient enrichment. By visualizing the correlations between different taxa, it appears there are 3 main taxa that have strong positive relationships with each other: Diaphanasoma sp., cyclopoid copepods, and Bosmina sp. With this correlation, I believe the biomass response of zooplankton to nutrent enrichment could be attributed to the increase in these 3 taxa concurrently. I created barplots and ran one-way ANOVA for these three zooplankton species, which confirmed this hypothesis. Additionally, there is a negative correlation between Chydorus sp. with all other taxa, and this is most significant with the three aformentioned taxa. Therefore, this suggests that the presence of Chydorus sp. may negatively influence the biomass of other taxa, which may further contribute to the biomass accumulation of zooplankton in different nutrient treatments.

SUBMITTING YOUR WORKSHEET

Use Knitr to create a PDF of your completed **3.RStudio_Worksheet.Rmd** document, push the repo to GitHub, and create a pull request. Please make sure your updated repo include both the PDF and RMarkdown files.

This assignment is due on **Wednesday, January 18th, 2021 at 12:00 PM (noon)**.