3. Worksheet: Basic R

Mackenzie Caple; Z620: Quantitative Biodiversity, Indiana University

15 January, 2019

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. 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.
5. Before you leave the classroom today, it is *imperative* that you **push** this file to your GitHub repo.
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 16th, 2019 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.

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/mcaple/GitHub/QB2019_Caple/2.Worksheets/3.RStudio"
```

```
setwd("~/GitHub/QB2019_Caple/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.

```
#1)
5^3
```

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

```
#2)
pi*2^2
```

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

```
#3)
sqrt(2)*sin(pi/4)
```

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

```
#4)
log(3)
```

```
## [1] 1.098612
```

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(1, 1, 2, 3, 5)
w = x*14
(x+w)/15
```

```
## [1] 1 1 2 3 5
```

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(2, 2, 4, 6, 10)
k*x
```

```
## [1]  2  2  8 18 50
```

```
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)

sem = function(x){
  sd(na.omit(x))/sqrt(length(na.omit(x)))
}

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(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(5,8,2))
z = c(rnorm(5,25,10))
k = cbind(j,z)
```

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: 'rnorm' generates random values that follow a normal distribution– but default, they follow a distribution with mean = 0 and standard deviation = 1. The three arguments are: the number of random numbers generated, the mean of the distribution, and the standard deviation of the distribution.

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 = read.table("data/matrix.txt", header = TRUE)

n = t(m)
dim(n)
```

```
## [1] 5 9
```

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

Answer 2: 5 rows and 9 columns

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`.

```
m1 = m[ ,c(1:2,4:5)]
m2 = m[1:8, ]
```

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.

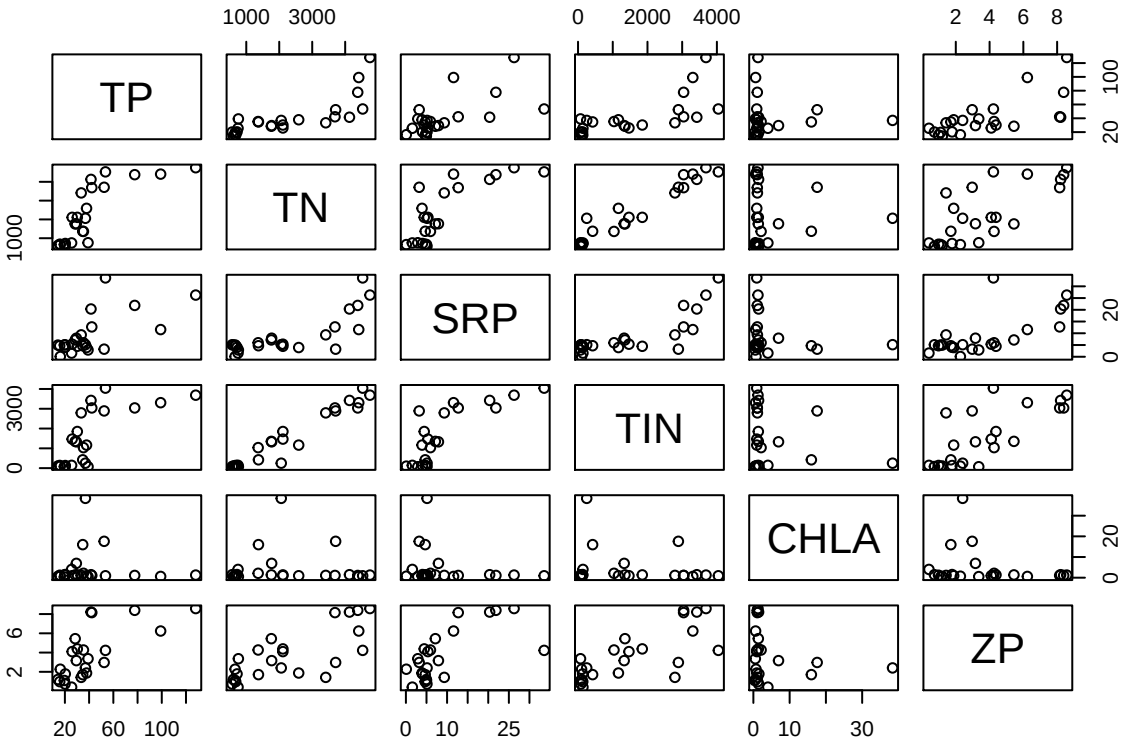
```
meso = read.table("data/zoop_nuts.txt", sep = "\t", header = TRUE)
str(meso)
```

```
## 'data.frame':    24 obs. of  8 variables:
##  $ TANK: int  34 14 23 16 21 5 25 27 30 28 ...
##  $ NUTS: Factor w/ 3 levels "H","L","M": 2 2 2 2 2 2 2 2 3 3 ...
##  $ 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
```

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

> Answer 3: Most nutrient pairs (combinations of TP, TN, SRP, TIN) have positive correlations of somewhat high magnitudes, meaning that these nutrients (total phosphorus, total nitrogen, soluble reactive phosphorus, and total inorganic nutrients), tend to increase or decrease with each other. All of those nutrient levels also have positive correlations with ZP, meaning that zooplankton biomass tends to increase at higher nutrient levels. However, CHLA (chlorophyll a concentration) has negative correlations with every other variable– although some of these correlations are weak, and none are as strong as the positive correlations between nutrients and zooplankton. Since CHLA is a proxy for algal biomass, this means that there may tend to be less algae in tanks with higher nutrients levels and higher levels of zooplankton.

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
```

```
## Warning: package 'psych' was built under R version 3.5.2
```

```
cor2 = corr.test(meso.num, method = "pearson", adjust = "BH")
cor3 = corr.test(meso.num, method = "kendall", 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
```

```
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: Yes, the results are definitely sensitive; correlation values tended to be lower in magnitude with a non-parametric method, and p-values tended to increase. Some of the correlations with chlorophyll a concentration even switched sign, though the magnitude remained fairly low, and the p-values very high. In general, non-parametric methods should be used when there is no reason to think that your data follow a specific type of distribution. I would not say that this data shows evidence for false discovery rate using the Pearson's method; although several p-values increased when corrected for multiple comparisons, they were already very large– no p-value changed from a significant to an insignificant level. However, false discovery rate is important to correct for, because when making a large number of comparisions, it is expected that some relationships will appear to be significantly correlated due to chance alone.

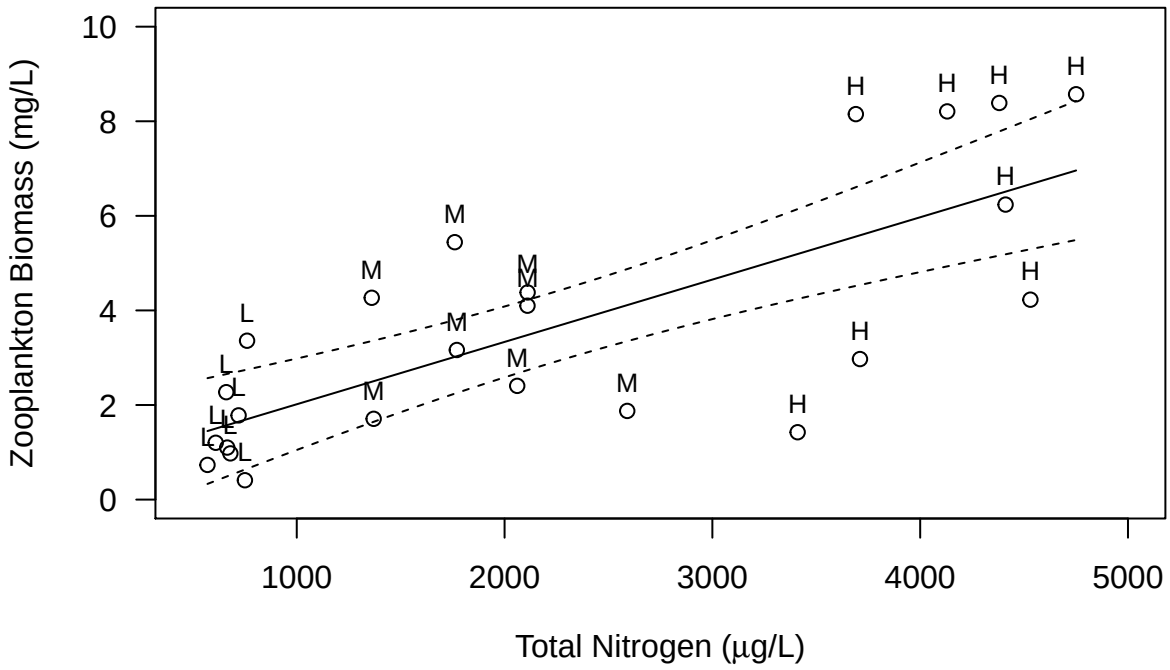
Linear Regression

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.

```
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
```

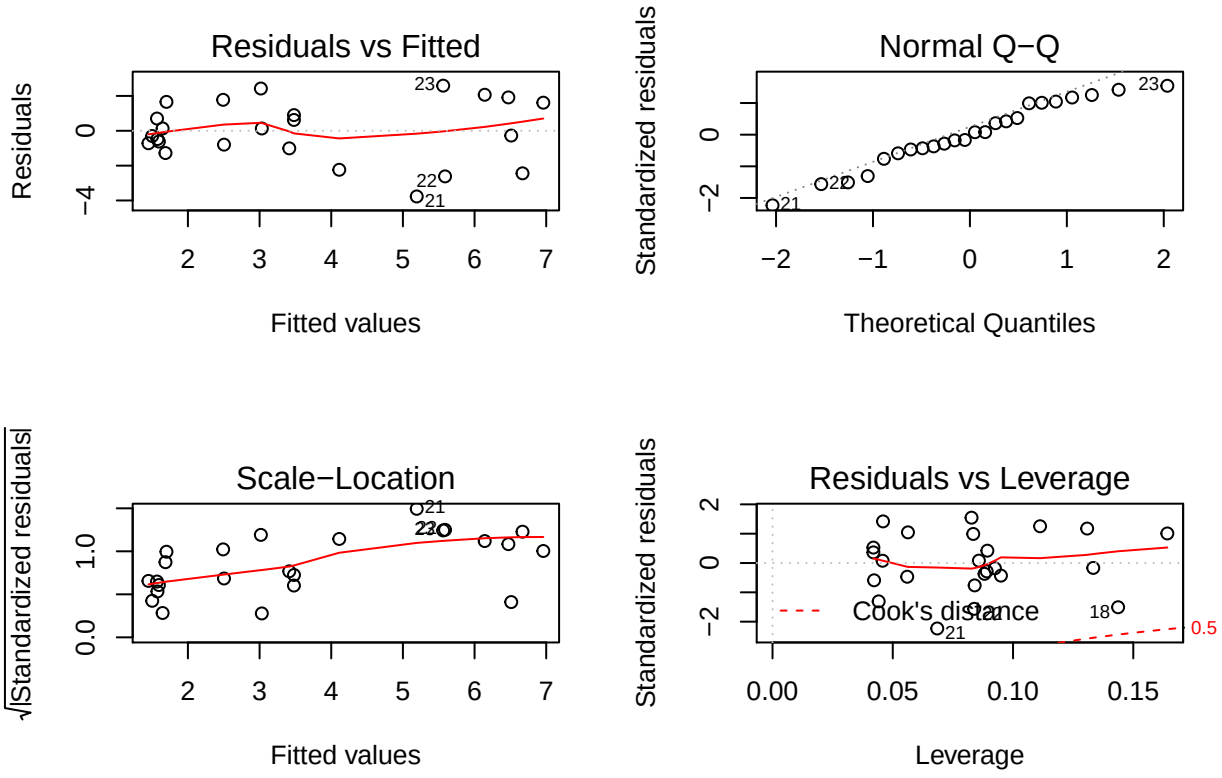
```
plot(meso$TN, meso$ZP, ylim = c(0,10), xlim = c(500,5000),
     xlab = expression(paste("Total Nitrogen (", mu, "g/L)")),
     ylab = "Zooplankton Biomass (mg/L)", las = 1)
text(meso$TN, meso$ZP, meso$NUTS, pos = 3, cex = 0.8)
newTN = seq(min(meso$TN), max(meso$TN), 10)
regline = predict(fitreg, newdata = data.frame(TN = newTN))
lines(newTN, regline)
conf95 = predict(fitreg, newdata = data.frame(TN = newTN),
                 interval = c("confidence"), level = 0.95, type = "response")
matlines(newTN, conf95[ , c("lwr", "upr")], type = "l", lty = 2, lwd = 1, col = "black")
```



Question 5: Interpret the results from the regression model

> Answer 5: Zooplankton biomass is highly significantly correlated with total nitrogen concentration, with a p-value of 1.91x10^-5. The residuals and square-root-standardized residuals seem to be reasonably randomly distributed around zero, the normal QQ plot is reasonably linear, and there are no points with a Cook's distance above (or even nearing) |1|, so a linear regression seems to be a reasonable way to analyze this relationship.

```
par(mfrow = c(2,2), mar = c(5.1, 4.1, 4.1, 2.1))
plot(fitreg)
```



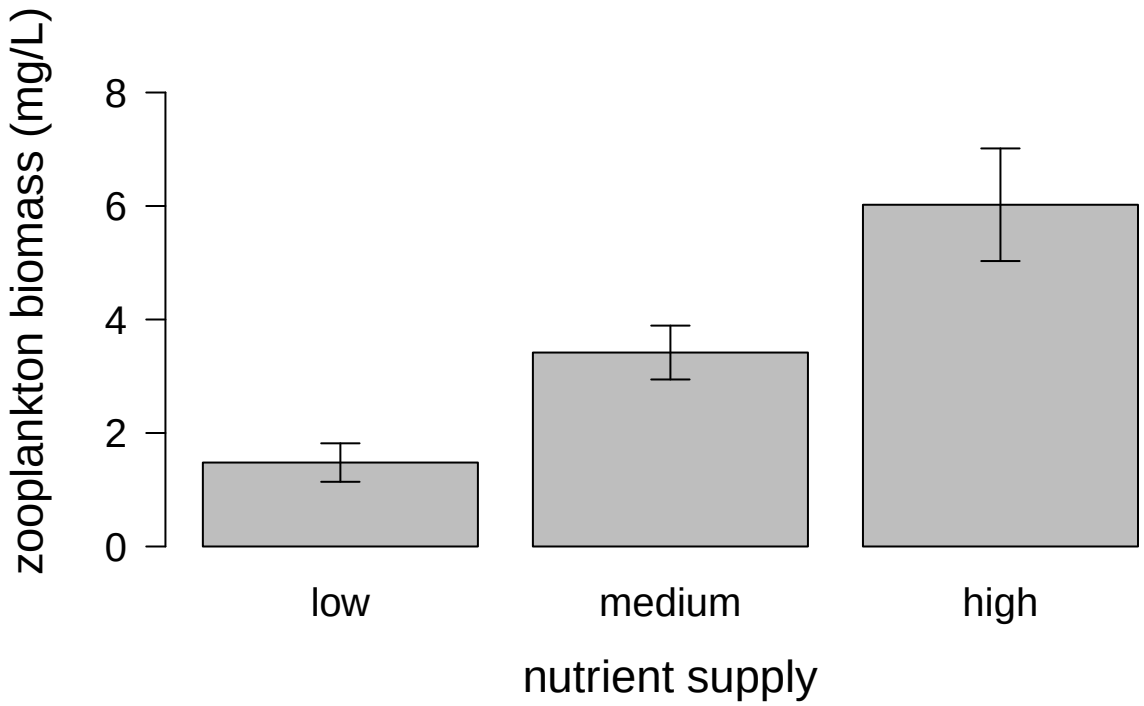
Analysis of Variance (ANOVA)

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 zoop.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

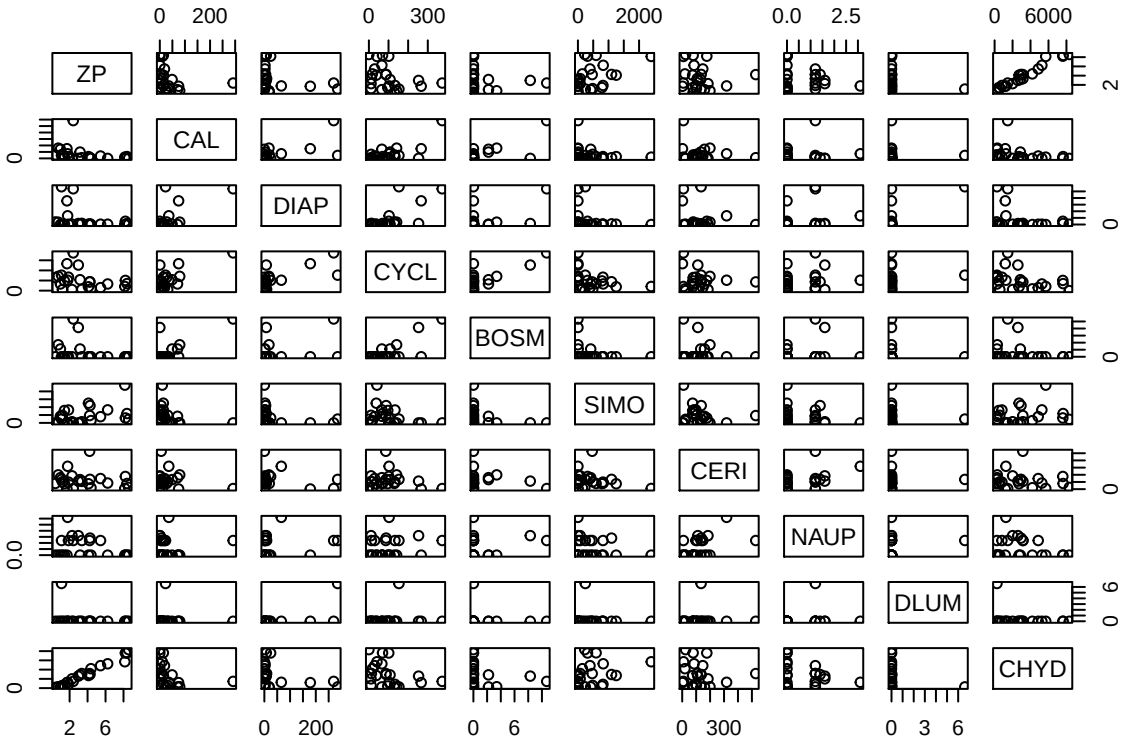
the total biomass (ZP) response to nutrient enrichment. Describe what you learned below in the "Answer" section and include appropriate code in the R chunk.

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

```
## 'data.frame':    24 obs. of  11 variables:
##  $ TANK: int  5 14 16 21 23 25 27 34 12 15 ...
##  $ NUTS: Factor w/ 3 levels "H","L","M": 2 2 2 2 2 2 2 3 3 ...
##  $ 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 ...
```

```
#combining the two dataframes based on tank ID
zp.all = merge(zoops, meso, by = "TANK")
sbsmat = zp.all[c(18,3:11)]

#simple pairwise comparisons and Pearson's correlations
pairs(sbsmat)
```




```
cor.zp = cor(sbsmat)
cor.zp
```

```
##              ZP         CAL       DIAP       CYCL        BOSM        SIMO
## ZP    1.0000000 -0.30662323 -0.2987959 -0.3554161 -0.21441404  0.43091831
## CAL  -0.3066232  1.00000000  0.6425998  0.7118997  0.72809861 -0.27145378
```

```
## DIAP -0.2987959  0.64259976  1.0000000  0.6943602  0.38105708 -0.28653648
## CYCL -0.3554161  0.71189969  0.6943602  1.0000000  0.74669146 -0.32477695
## BOSM -0.2144140  0.72809861  0.3810571  0.7466915  1.00000000 -0.30834317
## SIMO  0.4309183 -0.27145378 -0.2865365 -0.3247770 -0.30834317  1.00000000
## CERI -0.1408163 -0.19120238 -0.1723296 -0.1321577 -0.14137566 -0.18254802
## NAUP -0.2436860  0.05768141  0.2167634  0.1855902  0.17889950 -0.23678921
## DLUM -0.2068518 -0.03354653  0.6366939  0.1252018 -0.08630324 -0.07658909
## CHYD  0.9811264 -0.32170115 -0.3135602 -0.3685304 -0.20621888  0.26236337
##             CERI        NAUP        DLUM       CHYD
## ZP   -0.14081629 -0.24368598 -0.20685182  0.9811264
## CAL  -0.19120238  0.05768141 -0.03354653 -0.3217012
## DIAP -0.17232956  0.21676337  0.63669394 -0.3135602
## CYCL -0.13215770  0.18559022  0.12520183 -0.3685304
## BOSM -0.14137566  0.17889950 -0.08630324 -0.2062189
## SIMO -0.18254802 -0.23678921 -0.07658909  0.2623634
## CERI  1.00000000  0.47454001  0.02021705 -0.1354263
## NAUP  0.47454001  1.00000000  0.14753935 -0.2376803
## DLUM  0.02021705  0.14753935  1.00000000 -0.2240296
## CHYD -0.13542629 -0.23768026 -0.22402962  1.0000000
```

Answer 6: I created a site-by-species matrix by merging the two dataframes based on tank ID, and then indexed the new dataframe so it only included the ZP data and the biomass of each taxa. I ran a simple Pearson's correlation (ignoring all but the first row/column of the output) to see how each taxon contributed to total biomass. The correlations show that most of the positive response of zooplankton biomass to nutrient increase is driven by just one taxon, CHYD, which is very highly correlated with ZP. The taxon SIMO also contributed to the positive trend. All other taxa had negative correlations with total zooplankton biomass, though they were fairly weak correlations.

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 16th, 2015 at 12:00 PM (noon)**.