Week 1 Assignment: Basic R

Venus Kuo; Z620: Quantitative Biodiversity, Indiana University

15 January, 2017

OVERVIEW

Week 1 Assignment introduces some of the basic features of the R computing environment (http://www.r-project.org). It is designed to be used along side your Week 1 Handout (hard copy). You will not be able to complete the exercise if you do not have your handout.

Directions:

1. Change "Student Name" on line 3 (above) with your name.
2. Complete as much of the assignment as possible during class; what you do not complete in class will need to be done on your own outside of 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 exercise.
4. Be sure to **answer the questions** in this assignment document. 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 assignment, **Knit** the text and code into a single PDF file. Basically, just press the `Knit` button in the RStudio scripting panel. This will save the PDF output in your Week1 folder.
7. After Knitting, please submit the completed exercise by making a **push** to your GitHub repo and then create a **pull request** via GitHub. Your pull request should include this file (*Week1_Assignment.Rmd*; with all code blocks filled out and questions answered) and the PDF output of `Knitr` (*Week1_Assignment.pdf*).

The completed exercise is due on **Wednesday, January 18$^{\text{th}}$, 2017 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 assignment.

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 Week1 folder.

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

```
## [1] "C:/Users/Venus/Github/QB2017_Kuo/Week1"
```

```
setwd("C:/Users/Venus/Github/QB2017_Kuo/Week1/")
```

3) USING R AS A CALCULATOR

To follow up on the Week 0 exercises, please calculate the following in the R code chunk below. Feel free to reference the Week 0 handout.

1) the volume of a cube with length, l, = 5.
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. 45Â°) and with hypotenuse length sqrt(2) (remember: sin(theta) = opposite/hypotenuse).
4) the log (base e) of your favorite number.

```
# 1
l <- 5
cube.vol <- l^3
cube.vol # 125 #
```

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

```
# 2
r <- 2
circle.area <- pi*r^2
circle.area # 12.56....#
```

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

```
# 3
theta <- pi/4
hypothenuse <- sqrt(2)
opposite <- hypothenuse/sin(theta)
opposite # 2 #
```

```
## [1] 2
```

```
# 4
Fav.log <- log(10)
Fav.log # 2.3....#
```

```
## [1] 2.302585
```

4) WORKING WITH VECTORS

To follow up on the Week 0 exercises, please perform the requested operations in the Rcode chunks below. Feel free to reference the Week 0 handout.

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.

```
# 1 #
x <- c(1:5)

# 2 #
w <- x*14
w
```

```
## [1] 14 28 42 56 70
```

```
#3 #
c <- (x+w)/15
c # 1,2,3,4,5 #
```

```
## [1] 1 2 3 4 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`.

```
# 1 #
k <- vector(length = length(w))

# 2 #
k*x
```

```
## [1] 0 0 0 0 0
```

```
# 3 #
d <- c(w[1:3],k[1:4])
d
```

```
## [1] 14 28 42  0  0  0  0
```

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)

v <- na.omit(v)

max(v)
```

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

```
min(v)
```

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

```
sum(v)
```

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

```
sum(v)
```

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

```
mean(v)
```

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

```
median(v)
```

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

```
var(v)
```

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

```
sd(v)
```

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

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

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 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))
l <- matrix(cbind(z,j), nrow=5, ncol=2, byrow=FALSE)
dim(l)
```

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

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 random numbers under a normal distribution with means and standard deviation defined # # n = the number of observations, mean = vector of means, sd = vector of standard deviations #

In the R code chunk below, do the following: 1) Load `matrix.txt` from the Week1 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(n)
```

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

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

> Answer 2: # m is 10 columns and 5 rows and n is 5 columns and 10 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`.

```
m[,-3]
```

```
##       V1 V2 V4 V5
##  [1,]  8  1  6  1
##  [2,]  5  5  4  1
##  [3,]  2  5  3  3
##  [4,]  3  2  1  4
##  [5,]  9  9  1  2
##  [6,] 11  8  8  8
##  [7,]  2  2  8  5
##  [8,]  3  3  7  6
##  [9,]  5  5  3  6
## [10,]  6  5  2  2
```

```
m <- m[-10,]
```

Question 3: Describe what we just did in the last series of indexing steps.

> ***Answer 3***: # I indexed the m matrix to not include the 10th row and then I replaced it as the actual m matrix, removing the row #

6) BASIC DATA VISUALIZATION AND STATISTICAL ANALYSIS

Load Zooplankton Dataset

In the R code chunk below, do the following: 1) Load the zooplankton dataset from the Week1 data folder. 2) Display the structure of this data set.

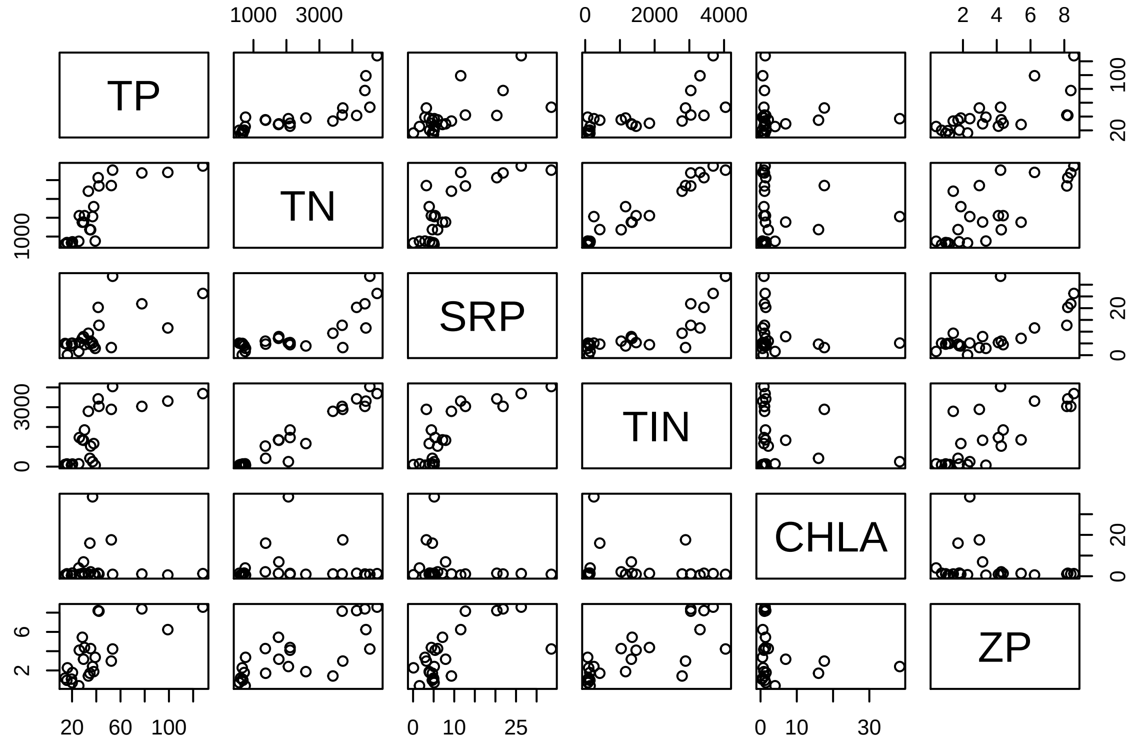
```
meso <- read.table("data/zoop_nuts.txt", sep="\t", header=TRUE)
str(meso)
```

```
## '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)
```



```r
cor1 <- cor(meso.num)
cor1
```

```
##               TP           TN        SRP        TIN         CHLA
## TP    1.00000000  0.786510407  0.6540957  0.7171143 -0.016659593
## TN    0.78651041  1.000000000  0.7841904  0.9689999 -0.004470263
## SRP   0.65409569  0.784190400  1.0000000  0.8009033 -0.189148017
## TIN   0.71711434  0.968999866  0.8009033  1.0000000 -0.156881463
## CHLA -0.01665959 -0.004470263 -0.1891480 -0.1568815  1.000000000
## ZP    0.69747649  0.756247384  0.6762947  0.7605629 -0.182599904
##              ZP
## TP    0.6974765
## TN    0.7562474
## SRP   0.6762947
## TIN   0.7605629
## CHLA -0.1825999
## ZP    1.0000000
```

Question 4: Describe some of the general features based on the visualization and correlation analysis above?

> Answer 4: # There is high correlation between factors like TN, ZP, and SRP with TIN. There are also negative correlations between factors like CHLA and all other nutrient (N/P) concentrations #

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.

```r
# install.packages("psych", repos = "https://cran.rstudio.com/")
require("psych")
```

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

```r
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
```

```r
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 5: Describe what you learned from `corr.test`. 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?

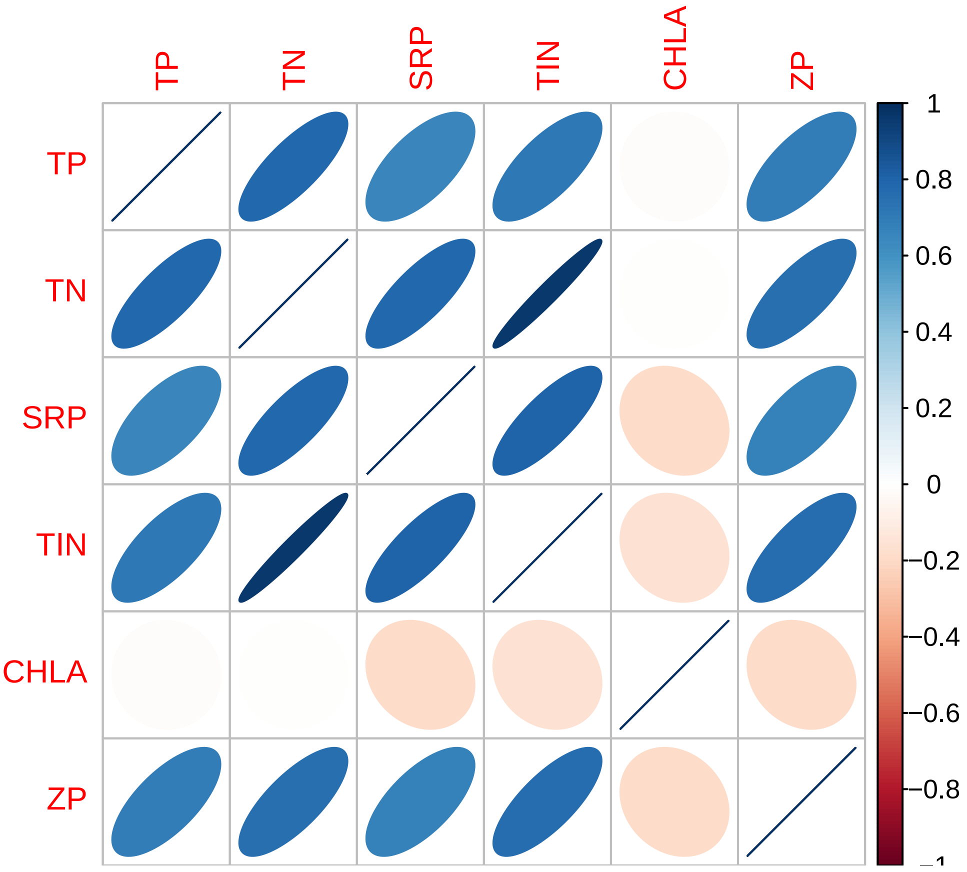
> ***Answer 5***: # The results seem to be fairly sensitive to parametric (Pearson's) method because the non-parametric method gave generally higher corrected p-values from the BH statement. Generally, you should use a non-parametric method when the dataset does not follow a particular, normal for example, distribution. It is possible to have correlations when considering many sub sets of data in a multiple comparisons, and those correlations may not be true (false discovery rate).

In the R code chunk below, use the `corrplot` function in the *corrplot* package to produce the ellipse correlation plot in the handout.

```r
# install.packages("corrplot", repos="http://cran.rstudio.com/")
require("corrplot")
```

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

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



```r
dev.off()
```

```
## null device
##           1
```

Linear Regression

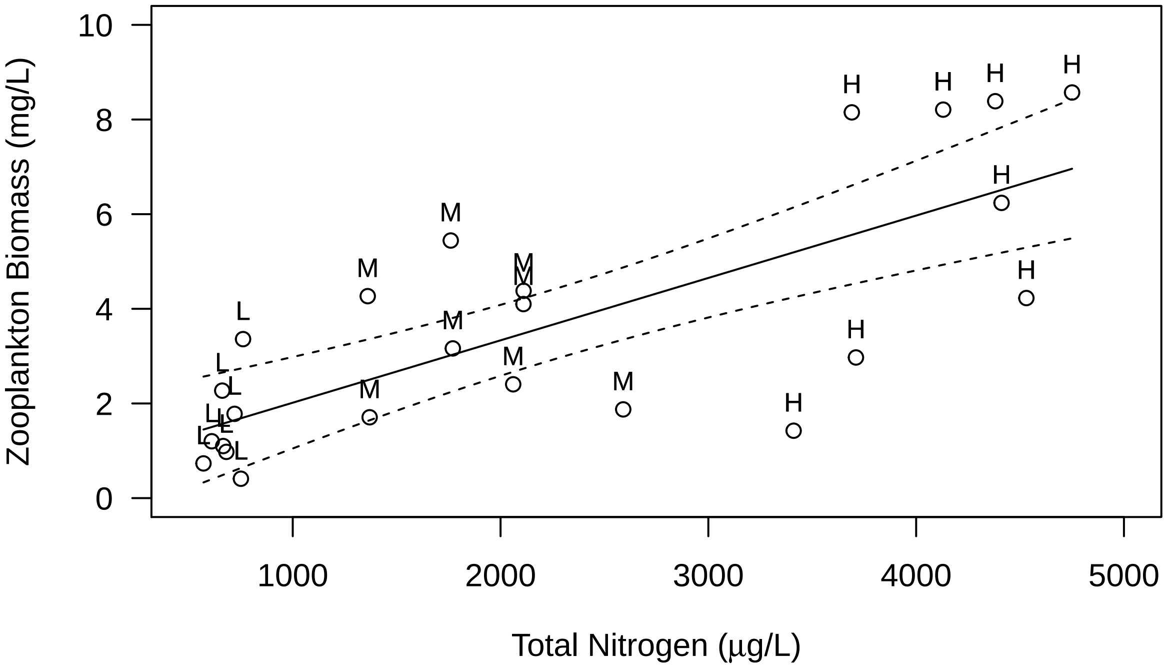
In the R code chunk below, do the following: 1) Conduct a linear regression analysis to test the relationship between total nitrogen (TN) and zooplankton biomass (ZP). 2) Examine the output of the regression analysis. 3) Produce a plot of this regression analysis including the following: categorically 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$NUTS, pos=3, cex=0.8)
text(meso$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 6: Interpret the results from the regression model

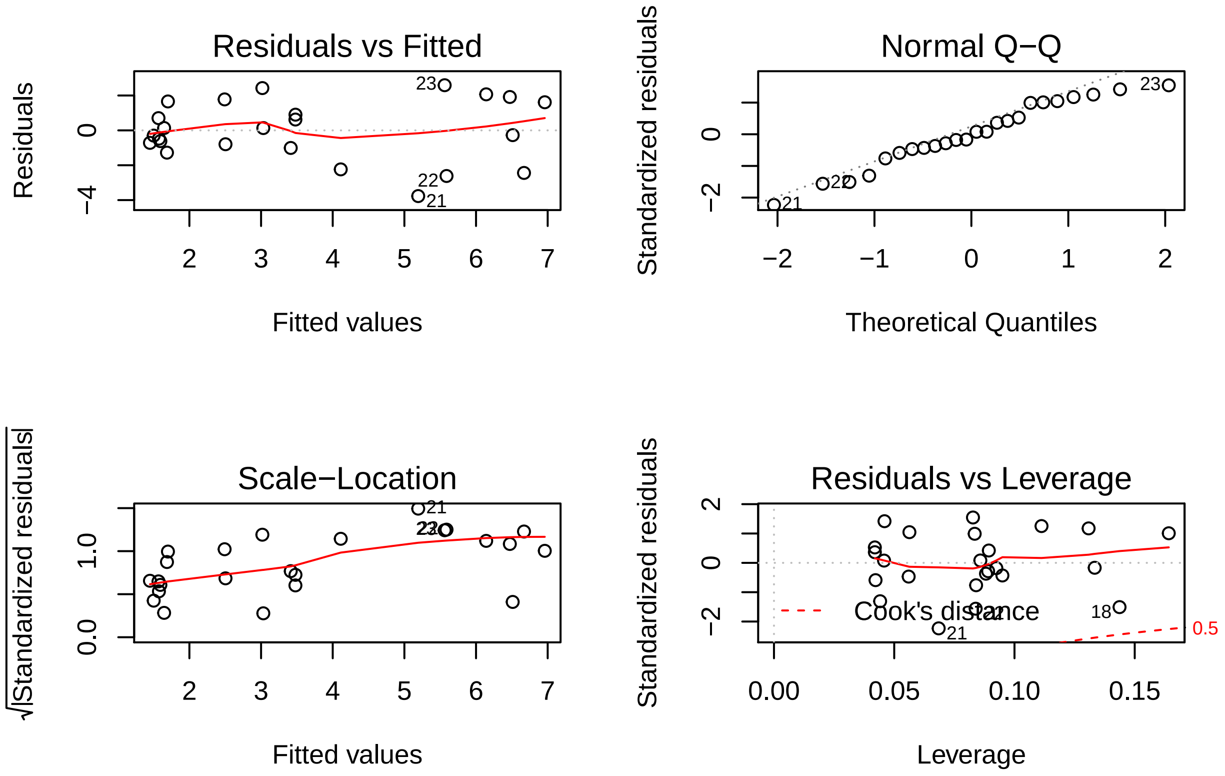
> ***Answer 6***: # The r-squared value is 0.5525 and the p-value is highly significant***. Zooplankton biomass is highly correlated with total nitrogen.

Question 7: Explain what the `predict()` function is doing in our analyses.

> ***Answer 7***: # The predict() is a function used for predicting the results of various model fitting functions.

Using the R code chunk below, use the code provided in the handout to determine if our data meet the assumptions of the linear regression analysis.

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



- Upper left: is there a random distribution of the residuals around zero (horizontal line)? #Yes
- Upper right: is there a reasonably linear relationship between standardized residuals and theoretical quantiles? Try `help(qqplot)` #Yes
- Bottom left: again, looking for a random distribution of sqrt(standardized residuals) # Not random distribution
- Bottom right: leverage indicates the influence of points; contours correspond with Cook's distance, where values > |1| are "suspicious"

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. 5) Use a Tukey's HSD to identify which treatments are different.

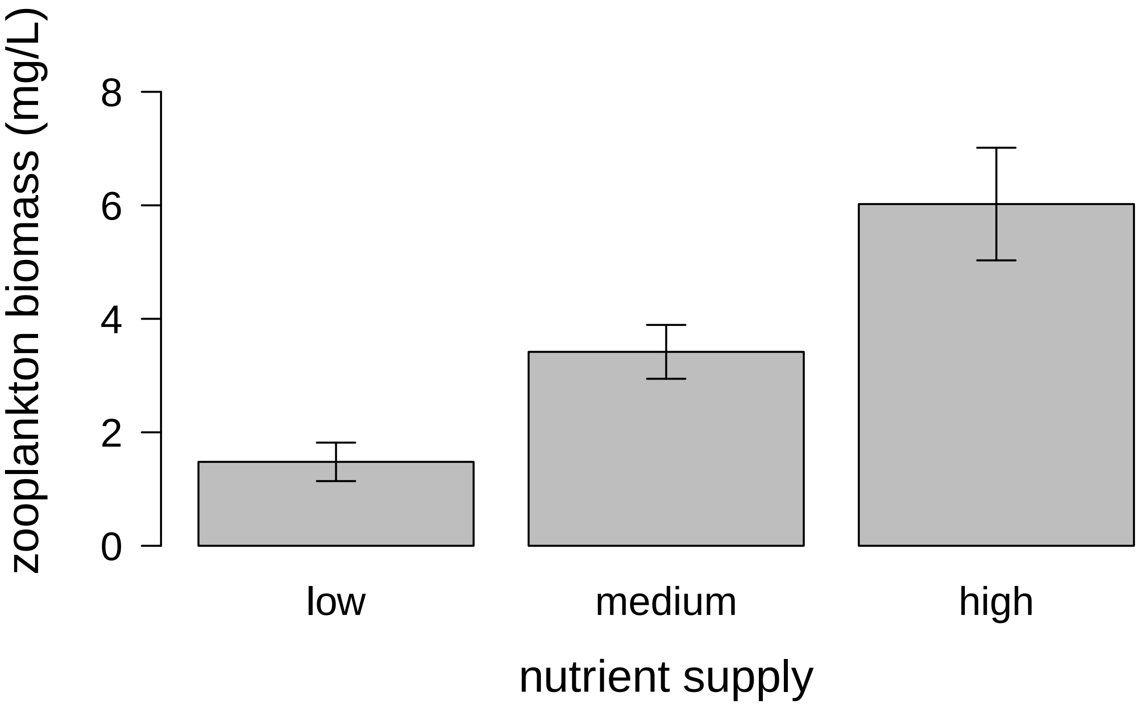
```
NUTS <-factor(meso$NUTS, levels=c('L', 'M', 'H'))

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
```

```
TukeyHSD(fitanova)
```

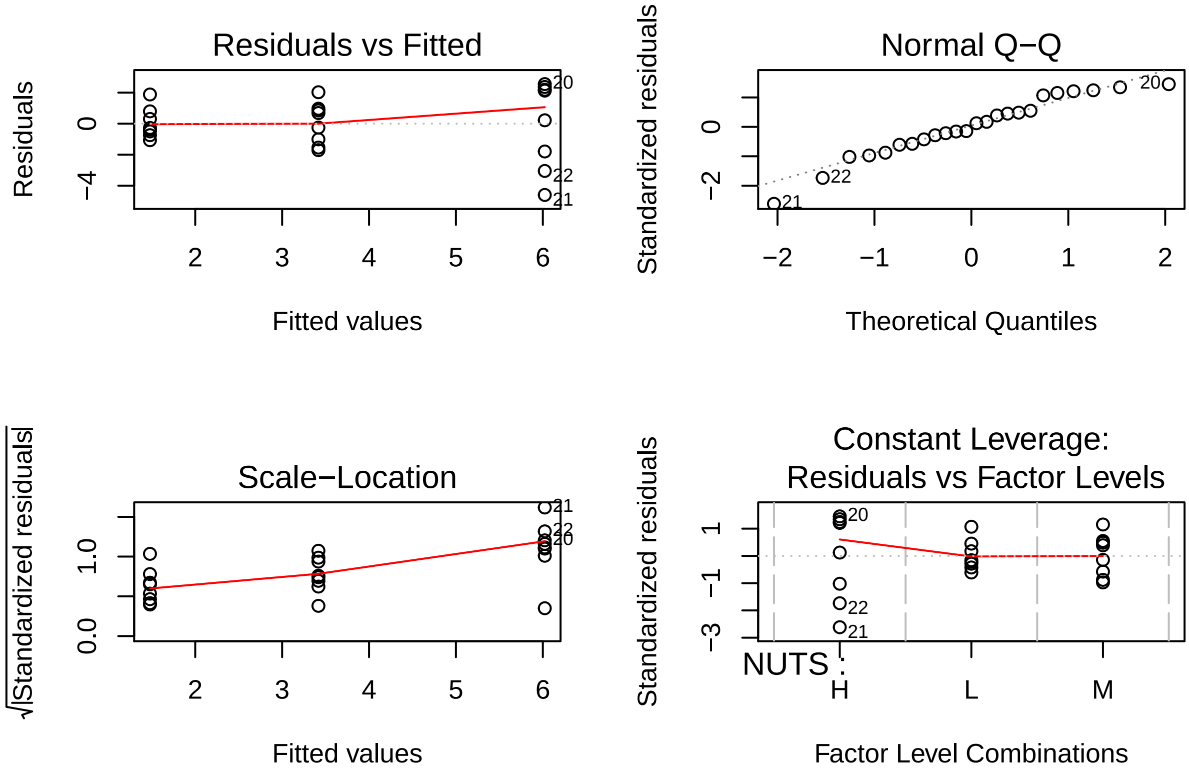
```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = ZP ~ NUTS, data = meso)
##
## $NUTS
##          diff        lwr        upr     p adj
## L-H -4.543175 -6.9115094 -2.1748406 0.0002512
## M-H -2.604550 -4.9728844 -0.2362156 0.0294932
## M-L  1.938625 -0.4297094  4.3069594 0.1220246
```

Question 8: How do you interpret the ANOVA results relative to the regression results? Do you have any concerns about this analysis?

> ***Answer 8***: # The nutrient supply to zoplankton biomass is significant in both ANOVA and regression tests, although the linear regression test gave a stronger significance. This ANOVA analysis seems to be less powerful of a test than the linear regression test.

Using the R code chunk below, use the diagnostic code provided in the handout to determine if our data meet the assumptions of ANVOA (similar to regression).

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



SYNTHESIS: SITE-BY-SPECIES MATRIX

In the R code chunk below, load the zoop.txt dataset in your Week1 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 (Âµg/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 9: With the visualization and statistical tools that we learned about in the Week 1 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.

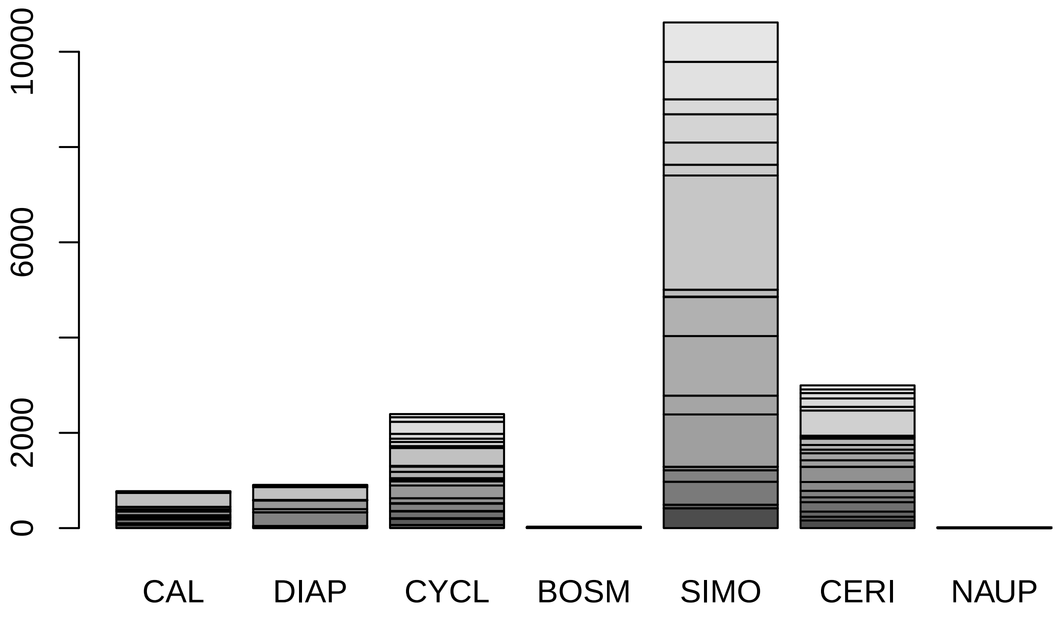
```
# Read zoop data set
zoop <- read.table("data/zoops.txt", sep="\t", header=TRUE)
# Making a site-by-species matrix
zoop.num <- zoop[,3:9]
# Summing across rows
Row.sums <- rowSums(zoop.num)
Row.sums
```

```
##  [1]  716.4  255.3  208.6  440.8  626.3  839.0  273.0  526.1  521.9 1335.0
## [11]  592.7 1380.3 1067.1  302.9  182.7  955.3 2458.0  252.1 1086.3  763.7
## [21]  615.8  382.7  967.0  980.2
```

```
zoop.num.total <- cbind(zoop.num, Row.sums)
zoop.num.total
```

```
##      CAL  DIAP  CYCL BOSM   SIMO  CERI NAUP Row.sums
## 1   70.5   0.0  66.1  2.2  417.8 159.8  0.0    716.4
## 2   27.1  19.2 129.6  0.0    0.0  79.4  0.0    255.3
## 3    5.3   8.8  12.7  0.0   73.1 107.5  1.2    208.6
## 4   79.2  17.9 141.3  3.4    0.0 199.0  0.0    440.8
## 5   31.4   0.0  11.0  0.0  482.0 101.9  0.0    626.3
## 6   22.7 285.1 153.0  0.0  241.5 135.5  1.2    839.0
## 7    0.0   2.3  11.0  0.0   73.1 185.0  1.6    273.0
## 8   35.7  65.9 102.9  0.0    0.0 318.5  3.1    526.1
## 9   74.8 178.7 266.5  0.0    0.0   1.9  0.0    521.9
## 10   5.3   4.9  87.8  0.0 1099.2 136.4  1.4   1335.0
## 11  18.4   2.3  29.4  0.0  393.8 147.6  1.2    592.7
## 12  14.0   2.3  37.7  0.0 1251.5  74.8  0.0   1380.3
## 13  14.0   2.3 132.9  0.0  818.6  98.1  1.2   1067.1
## 14  48.8   2.3 107.9  2.2    9.0 132.7  0.0    302.9
## 15   0.0   0.0  17.7  0.0  145.3  19.7  0.0    182.7
## 16 292.0 269.5 373.4 10.7    0.0   8.5  1.2    955.3
## 17   9.7   0.0  41.1  0.0 2397.8   9.4  0.0   2458.0
## 18   0.0   2.3   0.0  0.0  225.5  24.3  0.0    252.1
## 19   5.3   0.0  86.2  0.0  465.9 527.7  1.2   1086.3
## 20  14.0   7.5  69.5  0.0  594.2  78.5  0.0    763.7
## 21   0.0  24.4 101.2  0.0  313.6 176.6  0.0    615.8
## 22   0.0   7.5 253.2  8.3    0.0 112.1  1.6    382.7
## 23   5.3   2.3  96.2  0.0  786.6  76.6  0.0    967.0
## 24   0.0   2.3  66.1  0.0  826.7  85.1  0.0    980.2
```

```r
barplot(as.matrix(zoop.num)) # Not sure if I did this right but it definitely shows SIMO dominating acr
```



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

# Taking the average of taxas across nutrient treatments
taxa.means <- by(zoop[,3:9], NUTS, FUN=colMeans)
taxa.means <- aggregate(zoop[,3:9],by=list(NUTS), mean)
taxa.means.num <- taxa.means[,2:8]
taxa.means
```

```
##   Group.1     CAL    DIAP     CYCL   BOSM     SIMO     CERI   NAUP
## 1       L 33.9875 49.9000  78.4500 0.7000 160.9375 160.8250 0.8875
## 2       M 58.4125 57.7875 131.6625 1.6125 464.6750  77.4625 0.6250
## 3       H  4.2875  5.7875  89.1875 1.0375 701.2875 136.2875 0.3500
```

****Answer 9**** # I am not sure if I did this correctly. By taking the sum of taxa biomass across sites, I was able tell that the SIMO taxa dominated the biomass across most sites, while BOSM and NAUP biomass were negligable. The same is true across nutrient levels, although biomass was more evenly distrubted among multiple taxas in the Low nutrient treatment, while SIMO dominated in high nutrient treatments.

SUBMITTING YOUR ASSIGNMENT

Use Knitr to create a PDF of your completed Week1_Assignment.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.

Unless otherwise noted, this assignment is due on **Wednesday, January 18th, 2015 at 12:00 PM (noon)**.