Today’s exercise will go over Spearman correlation and NMDS. NMDS is a type of ordination often used for environmental sampling There are different types of ordination that can be used depending on the question asked and the type of dataset.

Here are a few more links about ordination methods:

<https://ourcodingclub.github.io/2018/05/04/ordination.html>

<https://www.ohio.edu/plantbio/staff/mccarthy/multivariate/Ordination2010.pdf>

Before we get started, we need to set the working directory and put the data files in the designated working directory.

## Set Working Directory

## Make sure data file is located in working directory

#install.packages(c("devtools","Hmisc", "corrplot", "dplyr", "vegan"))

The c() allows you to combine several items at once

## Turn on packages:

library(ggplot2)

library(devtools)

library(vegan)

library(dplyr)

## import data via csv (make sure data is in working directory):

data.BM=read.csv("2018\_Beach\_Monitoring.csv", header = T)

There are several ways to import the data. We can import it through the GUI icon in the working environment or use the code

## check the dimensions of dataset

## Data should have 517 rows, 22 columns

dim(data.BM)

## Look at data.BM dataset, notice there are null or blanks

## remove null data/black cells, and dataset should have 40 observations:

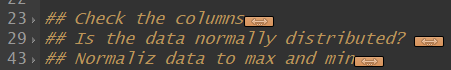
data.BM<-na.omit(data.BM)

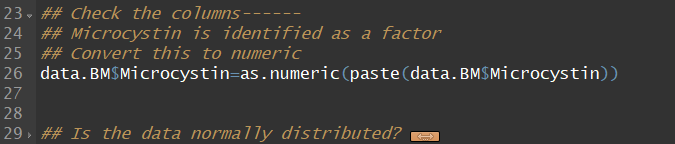
dim(data.BM) #Check dimensions

There are several ways to remove N/A cells and it depends on where the code is applied. If “na.omit()” does not work, check the package/function using “??package name” to see how to omit N/A values

## Check the columns------

Throughout this exercise, there will be “---------“ and you will see and drop arrow/triangle next to the numbered lines. The line associated with “-----“ will turn into an outline and everything else below this will be part of the outline. This is useful when the script has hundreds or thousands of lines. For example:





## Microcystin is identified as a factor

## Convert this to numeric

data.BM$Microcystin=as.numeric(paste(data.BM$Microcystin))

## Is the data normally distributed? ------

## null hypothesis: data is normally distributed,

## if p-value <0.05, then data is not normally distributed

## In such cases, certain statistical methods may not be appropriate

## and can lead to inaccurate interpretations

shapiro.test(data.BM$TDFe)

shapiro.test(data.BM$Microcystin)

## We can also test normality using K-S test

ks.test(data.BM$TDFe, "pnorm", 1, 2) #mean =1, 2= std dev

## Can also visually check for normal distribution

qqPlot(data.BM$Microcystin) #you can also try other variables

Environmental samples are rarely normally distributed. This is because these samples vary in space. For example, lake geochemical data are strongly influence by topography. Lakes with large watershed tend to have high concentrations. Lake surface area is also another factor, as smaller lakes tend to have higher concentrations.

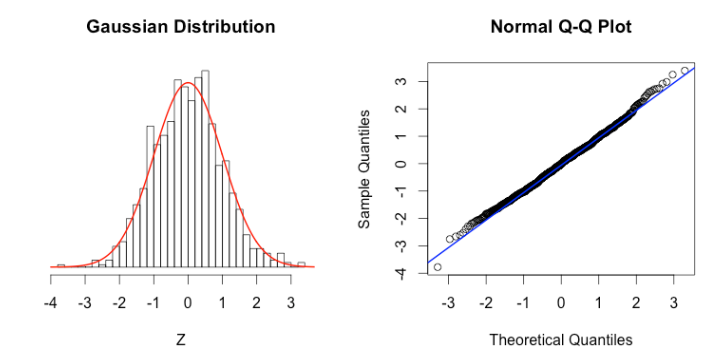


Figure . Normal distribution using QQ-plot

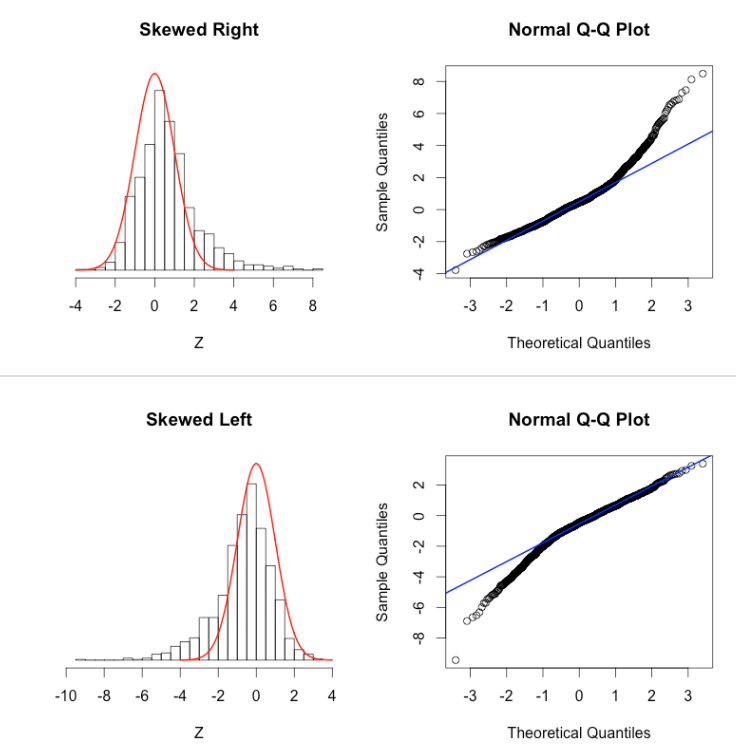


Figure . Not normal distribution QQ-plot. This indicates data needs to be transformed and then checked for normality

## Scale data to max and min-----

## Data values vary in range (from 0 to 100+)

## The function to normalize data is (x - min(x))/(max(x) - min(x))

## First, scale env't dataset using apply() and converting to a dataframe

data\_norm <- as.data.frame(apply(data.BM[, 9:19], 2,

function(x)(x - min(x))/(max(x)-min(x))))

Before we start, we need to understand our data. The values within this exercise range from 0 to 80+, therefore we need to scale the dataset from 0 to 1 for each variable so that the min value is 0 and max value is 1. This gives all variables equal footing (or whatever you want to call it). If this code does not work, exit out of R and restart it and run the codes again.

### Research Question-----

## What environmental conditions associate with microcystin?

## Compute p-value and Spearman correlation-------

## Often, Spearman correlation is used for environmental samples

## Because environmental samples are not normally distributed

## Pearson correlation is used when dataset is normally distributed

library(Hmisc) #use this package for correlations

res.corr<-rcorr(as.matrix(data\_norm[,c(1:11)], type = "spearman"))

res.corr #See results

[,c(1:11)] means we want to use columns 1:11. If we want to look at x rows, then the code can be changed to [c(1:200),] to look at rows 1 to 200. If you see N/A, then the data\_norm has N/A values.

## Graphing correlation plot------

## Extract p-value from results:

pval<-res.corr$P

## Extract Pearson correlation coefficient from results:

rval<-res.corr$r

## Plot Pearson correlation with p-value:

library(corrplot) #use this package to plot correlation

corr.plot<-corrplot(rval, type = "upper", order ="FPC",

p.mat = pval, sig.level = 0.05,

tl.col = "black", tl.srt = 45, tl.cex = 1.2,

cl.cex = 1.2, insig = "blank")

## tl = text label, cl = color label, col = columns, .cex magnifies the text size

## p-mat = p-value, type ??corrplot for more info

The insig = “blank” can be removed or replaced with insig = “p-value”. Check out the vignettes for inspiration!

## plot correlation using ggplot

## Note: this is a bit more complicated with coding

## but feel free to explore

There are numerous ways to plot a visually aesthetically pleasing graph/chart without information overload. If using ggplot, then the reshape2 package is needed. This package rearranges (restructure) our dataset in such a way that can be easily plotted.

#install.packages("reshape2")

library(reshape2)

res.corr2<-round(cor(data\_norm[,c(1:11)], method = "spearman"),2) #round 2 decimal places

melted\_corr<-melt(res.corr2) #rthe melt() will earrange correlation data for plotting

P1<-ggplot(data = melted\_corr, aes(x = Var1, y = Var2, fill = value)) +

geom\_tile() + scale\_fill\_gradient(low = "blue", high = "red")

P1

## Note that, a correlation matrix has redundant information.

## We'll use the function below to grab the upper portion of

## matric and set half of it to NA

get\_upper\_tri<-function(res.corr2){

res.corr2[lower.tri(res.corr2)]<- NA

return(res.corr2)

}

upper\_tri<-get\_upper\_tri(res.corr2) #grabs the upper portion of matrix

## heatmap of upper portion of matrix

P2<-ggplot(data = melt(upper\_tri, na.rm = TRUE), aes(x = Var1, y = Var2, fill = value)) +

geom\_tile(colour = "white") +

scale\_fill\_gradient2(low = "red", high = "blue", mid = "white",

midpoint = 0, limit = c(-1,1), space = "Lab",

name = "Spearman Rank Correlation") + #correlation coefficients range from -1 to 1.

theme(panel.grid.major = element\_blank (),panel.grid.minor = element\_blank(),

panel.background = element\_rect (fill = "white"),text = element\_text(size = 14),

axis.line = element\_line(colour = "black"),

axis.title = element\_text(size = 14),

legend.text = element\_text(size = 14),

axis.text.y = element\_text(size = 14),

axis.text.x = element\_text(angle = 45, size = 14, hjust = 1)) +

coord\_fixed() #ensures that one unit on x-axis is same as y-axis

P2

## Add r value to heatmap

P3<-P2 +

geom\_text(aes(Var1, Var2, label = value), color = "black", size = 4)

P3

##Research Question ------

##What environmental conditions associate with microcystin?

## NMDS uses rank orders to observe how species or composition

## change from one community to the next

## k can be increased if stress is too high

library(vegan)

mds.out=metaMDS(data\_norm[,1:11],distance="bray", k=3)

High stress can be reduced by increasing the k value (# dimensions). Low stress is good!!!

Stress < 0.05 is excellent

Stress < 0.1 is great

Stress < 0.2 is OK

Stress < 0.3 is bad ……..but stress close to 0 means there is trouble in the data

## Put NMDS output into a dataframe------

data.sm<-as.data.frame(scores(mds.out))

data.sm

## Add columns from data.BM to mds.out dataframe

data.sm$Date=as.character(data.BM$Date)

data.sm$Site<-as.character(data.BM$Site) #alternative way to add

data.sm$Month=as.character(data.BM$Month)

## Plot NMDS using ggplot-----

P4<-ggplot(data=data.sm, aes(x=NMDS1, y=NMDS2, color=Month)) +

geom\_point(size=3) + xlim(-0.6, 0.6) + ylim(-0.6, 0.6)+

geom\_hline(yintercept=0.0, colour="grey", lty=2)+

geom\_vline(xintercept=0.0, colour="grey",lty=2) +

theme(legend.background = element\_rect(fill="white", size=0.3,

linetype="solid", colour="black"),

panel.background = element\_blank(), panel.grid.major = element\_blank(),

panel.grid.minor = element\_blank(), axis.line = element\_line(colour = "black"),

axis.text=element\_text(size=14), axis.title = element\_text(size = 14))

P4

## adding arrows

env.monitoring=envfit(mds.out, data.BM[,9:19], perm=1001)

env.monitoring

arrow.monitoring=as.data.frame(scores(env.monitoring,display="vectors"))

arrow.monitoring

arrow.monitoring=cbind(arrow.monitoring,Species=rownames(arrow.monitoring))

plot(arrow.monitoring)

P5<-ggplot(data=data.sm, aes(x=NMDS1, y=NMDS2)) +

geom\_point(aes(colour=Month), size=3) +

xlim(-0.8, 0.8) + ylim(-0.8, 0.8) +

geom\_hline(yintercept=0.0, colour="grey", lty=2) +

geom\_vline(xintercept=0.0, colour="grey",lty=2) +

theme\_bw() + scale\_color\_brewer(palette="Set1") + theme(legend.position = c(0.9, 0.7)) +

theme(legend.background = element\_rect(fill="white", size=0.3, linetype="solid", colour="black"))+

geom\_segment(data=arrow.monitoring, aes(x=0, xend=2\*NMDS1, y=0, yend=2\*NMDS2),

arrow=arrow(length=unit(0.25,"cm")))+

geom\_text(data=arrow.monitoring, aes(x=2.2\*NMDS1, y=2.2\*NMDS2, label=Species), size=5)

P5

Change **plot titles** by using the functions ggtitle(), xlab() and ylab() or labs(). There are many ways to navigate through ggplot. For example:

P6<- P5 + labs(title =”hdhfbhjdf”, x = “dfdgf”, y = “dhgfbd”)

--or—

P6<- P5 + ggtitle(“dhjfgdb”) + xlab (“dhfbdj”) + ylab(“ghjdbgj”)

For more and more inspiration:

<http://www.sthda.com/english/wiki/ggplot2-essentials>

<http://www.sthda.com/english/wiki/ggplot2-title-main-axis-and-legend-titles>

## MRPP & ANOSIM

ANOSIM (ANalysis Of SIMilarity) is a method that tests whether two or more groups of samples are significantly different. This is often used to align the NMDS results. You can specify a category in the metadata to separate samples into groups and then test whether there are significant differences between those groups.  Since ANOSIM is nonparametric, statistical significance is determined through permutations. We can increase the number of permutations to overcome the not-normal distribution for environmental samples

envt=as.matrix(data\_norm[,1:11])

Microcystin=as.numeric(data\_norm[,1])

mrpp(envt, group=Microcystin, distance="bray")

anosim(envt, grouping=Microcystin, permutations=999, distance="bray")

An R value close to "1.0" suggests dissimilarity between groups while an R value close to "0" suggests an even distribution of high and low ranks within and between groups. R values below "0" suggest that dissimilarities are greater within groups than between groups.