Bacterioplankton response to salinity (aka 2015 CSI Dispersal Experiment)

Alex Stucy, Jo Werba, Mike McCoy, Ariane Peralta

Last updated on 31 May, 2017

```r
rm(list = ls())
setwd("~/GitHub/CSI_Dispersal/analyses")
```

```r
#Set Std Err and Conf Int
se <- function(x, ...) {
sd(x, na.rm = TRUE)/sqrt(length(na.omit(x)))
}
ci <- function(x, ...) {
1.96 * sd(x, na.rm = TRUE)
}
```

```r
#Set Source R Tools
source("../bin/DiversityFunctions.R")
source("../bin/MothurTools.R")
```

```
## Loading required package: reshape
```

```
## Warning: package 'reshape' was built under R version 3.2.5
```

```r
#load required packages
require("vegan")
```

```
## Loading required package: vegan
```

```
## Warning: package 'vegan' was built under R version 3.2.5
```

```
## Loading required package: permute
```

```
## Warning: package 'permute' was built under R version 3.2.5
```

```
## Loading required package: lattice
```

```
## Warning: package 'lattice' was built under R version 3.2.5
```

```
## This is vegan 2.4-2
```

```r
require("dplyr")
```

```
## Loading required package: dplyr
```

```
## Warning: package 'dplyr' was built under R version 3.2.5
```

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

```
## The following object is masked from 'package:reshape':
##
##     rename
```

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

```r
require("nlme")
```

```
## Loading required package: nlme
```

```
## Warning: package 'nlme' was built under R version 3.2.5
```

```
##
## Attaching package: 'nlme'
```

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

```r
require("reshape")
require("BiodiversityR")
```

```
## Loading required package: BiodiversityR
```

```
## Warning: package 'BiodiversityR' was built under R version 3.2.5
```

```
## Loading required package: tcltk
```

```r
require("ecodist")
```

```
## Loading required package: ecodist
```

```
##
## Attaching package: 'ecodist'
```

```
## The following object is masked from 'package:vegan':
##
##     mantel
```

```r
require("ggplot2")
```

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 3.2.5
```

```r
require("ade4")
```

```
## Loading required package: ade4
```

```
## Warning: package 'ade4' was built under R version 3.2.5
```

```
##
## Attaching package: 'ade4'
```

```
## The following object is masked from 'package:vegan':
##
##     cca
```

```r
require("png")
```

```
## Loading required package: png
```

```r
#load design file - this is the real design file
design <- read.csv("../data/CSI_Design_ENV_NoSourceTanks.csv", row.names=1)
head(design)
```

```
##        Field_ID    Date Date2 Replicate Treatment Dispersal Salinity
## CSI004        4 6/11/15     0         A         4         3        0
## CSI005        5 6/11/15     0         A         5         2        5
## CSI006        6 6/11/15     0         A         6         3        5
## CSI007        7 6/11/15     0         A         7         2        9
## CSI008        8 6/11/15     0         A         8         3        9
## CSI009        9 6/11/15     0         A         9         2       13
##        Salinity_real NH4um NO3um PO4um Maple_dmass Spartina_dmass
## CSI004          0.30  0.52  0.00  0.00        2.66           2.70
## CSI005          5.72  0.92  0.00  7.23        2.19           1.56
## CSI006          6.04  0.73  0.97  5.09        2.04           1.82
## CSI007          9.57  1.11  0.00  6.23        1.28           1.14
## CSI008         10.60  1.02  0.00  7.27        1.39           0.47
## CSI009         15.32  0.85  0.00  8.37        1.07           0.83
##        Phrag_dmass    Sample..
## CSI004        8.25 A_4_11JUN15
## CSI005        5.03 A_5_11JUN15
## CSI006        5.01 A_6_11JUN15
## CSI007        5.20 A_7_11JUN15
## CSI008        4.01 A_8_11JUN15
## CSI009        0.79 A_9_11JUN15
```

```
str(design)
```

```
## 'data.frame':    93 obs. of  15 variables:
##  $ Field_ID      : int  4 5 6 7 8 9 10 11 15 16 ...
##  $ Date          : Factor w/ 3 levels "6/11/15","6/29/15",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Date2         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Replicate     : Factor w/ 4 levels "A","B","C","D": 1 1 1 1 1 1 1 1 2 2 ...
##  $ Treatment     : int  4 5 6 7 8 9 10 11 4 5 ...
##  $ Dispersal     : int  3 2 3 2 3 2 3 2 3 2 ...
##  $ Salinity      : int  0 5 5 9 9 13 13 0 0 5 ...
##  $ Salinity_real : num  0.3 5.72 6.04 9.57 10.6 ...
##  $ NH4um         : num  0.52 0.92 0.73 1.11 1.02 0.85 1.33 0.92 0.82 0.81 ...
##  $ NO3um         : num  0 0 0.97 0 0 0 0 0 0.81 0 ...
##  $ PO4um         : num  0 7.23 5.09 6.23 7.27 8.37 8.36 0 7.07 4.14 ...
##  $ Maple_dmass   : num  2.66 2.19 2.04 1.28 1.39 1.07 2.23 2.52 1.7 1.85 ...
##  $ Spartina_dmass: num  2.7 1.56 1.82 1.14 0.47 0.83 1.66 1.85 1.75 1.93 ...
##  $ Phrag_dmass   : num  8.25 5.03 5.01 5.2 4.01 0.79 5.48 5.2 5.17 6.52 ...
##  $ Sample..      : Factor w/ 31 levels "A_10_11JUN15",..: 3 4 5 6 7 8 1 2 11 12 ...
```

Microbial Data

```
#load design file - this is the design file associated with microbes and needs to get subset w/o mock
design_crobes <- read.csv("../data/design_CSI.csv", row.names=1)
design_crobes <- design_crobes[-c(grep("mock community", design_crobes$CSI_ID)), ]
dim(design_crobes)
```

```
## [1] 129   7
```

```
# Import OTU data
# Import Raw Data
CSIdata.in <- read.otu("../data/CSI.shared")
```

```r
dim(CSIdata.in)
```

```
## [1]   128 19954
```

```r
# Removing Extra Site in Design site = CSI041
missing <- setdiff(rownames(design_crobes), rownames(CSIdata.in))
design_crobes <- design_crobes[-(which(rownames(design_crobes) == missing)), ]
dim(design_crobes)
```

```
## [1] 128   7
```

```r
design <- design[-(which(rownames(design) == missing)), ]
dim(design) #92,15
```

```
## [1] 92 15
```

```r
# Identify source tanks where Number = 1, 2, 3
temp <- rownames(design_crobes[which(design_crobes$Number %in% c(1, 2, 3)), ])

# Remove sampeles from tank numbers 1, 2, 3
bac.design <- design_crobes[-(which(rownames(design_crobes) %in% temp)), ]
design_crobes2 <- droplevels(bac.design)
dim(design_crobes2)
```

```
## [1] 92  7
```

```r
# Remove source tanks where Number = 1, 2, 3
CSIdata.in2 <- CSIdata.in[-(which(rownames(CSIdata.in) %in% temp)), ]
dim(CSIdata.in2)
```

```
## [1]    92 19954
```

```r
# Remove OTUs with less than two occurences across all sites
CSIdat.a <- CSIdata.in2[, which(colSums(CSIdata.in2) >= 2)]
dim(CSIdat.a)
```

```
## [1]    92 11620
```

```r
# Rarefy Abundances (min abundance is ___ after removing samples <10000) - need to fix
#aa <- (rowSums(CSIdat.a))
#aa
#CSI.r <- rrarefy(CSIdat.a, 13000)

#removed low samples (CSI101 had 75 reads)
CSIdat.b <- CSIdat.a[which(rowSums(CSIdat.a) >= 13000), ]
dim(CSIdat.b)
```

```
## [1]    91 11620
```

```r
# Odd sites in bacterial composition data (CS101) and remove CS041 in design file
odd.sites <- c("CSI101")

CSIdata.in3 <- CSIdat.b[setdiff(rownames(CSIdat.b), odd.sites), ]
design2 <- design[setdiff(rownames(design), odd.sites), ]

all.equal(rownames(design2), rownames(CSIdata.in3))
```

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

```r
#set treatments (salinity levels)
treatments1 <- as.factor(design2$Salinity)
levels(treatments1) <- c("0","5","9","13")
treatments2 <- as.factor(design2$Dispersal)
levels(treatments2) <- c("2","3")

# Make Presence Absence Matrix
CSIdataPA <- (CSIdata.in3 > 0) * 1

# Make Relative Abundence Matrices
CSIdataREL <- CSIdata.in3
for(i in 1:dim(CSIdata.in3)[1]){
  CSIdataREL[i,] <- CSIdata.in3[i,]/sum(CSIdata.in3[i,])
}
```

```r
#import taxonomy file - simplified name
CSI.tax1 <- read.tax(taxonomy = "../data/CSI.0.03.cons.taxonomy")
#create tax table .csv and export
write.table(CSI.tax1, file = "tax.csv", sep = ",",
col.names = NA)
```

```r
#bind design and bact files
newCSIdata <- cbind(design2,CSIdataREL)

#PERMANOVA
adonis = adonis(newCSIdata[,-c(1:15)] ~ Date2*Dispersal*Salinity, method = "bray", data = newCSIdata, p
adonis
```

```
##
## Call:
## adonis(formula = newCSIdata[, -c(1:15)] ~ Date2 * Dispersal *      Salinity, data = newCSIdata, permu
##
## Permutation: free
## Number of permutations: 1000
##
## Terms added sequentially (first to last)
##
##                           Df SumsOfSqs MeanSqs F.Model      R2   Pr(>F)
## Date2                      1     1.781  1.7813  5.5854 0.05232 0.000999 ***
## Dispersal                  1     0.248  0.2482  0.7782 0.00729 0.806194
## Salinity                   1     3.905  3.9052 12.2452 0.11471 0.000999 ***
## Date2:Dispersal            1     0.162  0.1617  0.5069 0.00475 1.000000
## Date2:Salinity             1     0.917  0.9173  2.8763 0.02695 0.001998 **
## Dispersal:Salinity         1     0.297  0.2965  0.9298 0.00871 0.520480
## Date2:Dispersal:Salinity   1     0.263  0.2631  0.8250 0.00773 0.702298
## Residuals                 83    26.470  0.3189         0.77754
## Total                     90    34.044                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
capture.output(adonis(newCSIdata[,-c(1:15)] ~ Date2*Dispersal*Salinity, method = "bray", data = newCSIda
```

```r
#matrix comparison - Is there a relationship between zooplankton and bacterial community? = YES Mantel

zoop <- read.csv("../data/zoop_CSI.csv", row.names=1)
```

```r
str(zoop)
```

```
## 'data.frame':    91 obs. of  33 variables:
##  $ CSI_ID         : Factor w/ 91 levels "ECU_CSI_004",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ ID             : int  4 5 6 7 8 9 10 11 15 16 ...
##  $ Date           : Factor w/ 3 levels "6/11/15","6/29/15",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Replicate      : Factor w/ 4 levels "A","B","C","D": 1 1 1 1 1 1 1 1 2 2 ...
##  $ Treatment      : int  4 5 6 7 8 9 10 11 4 5 ...
##  $ Dispersal      : int  3 2 3 2 3 2 3 2 3 2 ...
##  $ Calanoida      : int  14 268 47 60 287 397 74 0 0 36 ...
##  $ Chydoridae     : int  228 32 33 8 0 0 0 161 544 3 ...
##  $ Harpacticoida  : int  1 8 11 1 25 40 42 1 0 1 ...
##  $ Acartia        : int  0 0 0 0 0 10 0 0 0 0 ...
##  $ Ceriodaphnia   : int  38 0 0 0 0 0 0 8 41 0 ...
##  $ Ostracoda      : int  0 1 0 6 0 0 0 4 8 0 ...
##  $ Cyclopoida     : int  385 0 1 79 1 2 0 60 120 0 ...
##  $ Daphniidae     : int  1 0 0 0 0 0 0 2 1 0 ...
##  $ Bosminidae     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Isopoda        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Nauplius       : int  1 6 2 20 0 25 21 0 0 0 ...
##  $ Scapholeberis  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Simocephalus   : int  7 0 0 0 0 0 0 11 13 0 ...
##  $ Daphnia        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Rotifera       : int  0 7 0 0 0 0 0 0 0 0 ...
##  $ Unknown        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Water.mite     : int  0 0 0 0 0 0 3 0 0 0 ...
##  $ Sididae        : int  0 0 0 0 0 0 0 2 0 0 ...
##  $ Chironomidae   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Annelida       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Philopotamidae : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Caddisfly      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Jellyfish      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Mysidae        : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ Arthropoda     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Rhyacophiliidae: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Miridae        : int  0 0 0 0 0 0 0 0 0 0 ...
```

```r
dim(zoop)
```

```
## [1] 91 33
```

```r
#removed low samples
zoop.in <- zoop[,-c(1:6)]
dim(zoop.in)
```

```
## [1] 91 27
```

```r
#remove CSI026
odd.sites <- c("CSI026")
zoop.in2 <- zoop.in[setdiff(rownames(zoop.in), odd.sites), ]
CSIdata.in4 <- CSIdata.in3[setdiff(rownames(CSIdata.in3), odd.sites), ]

# Make Relative Abundence Matrices without CSI026
zoopREL <- zoop.in2
for(i in 1:dim(zoop.in2)[1]){
  zoopREL[i,] <- zoop.in2[i,]/sum(zoop.in2[i,])
}

CSIdataREL2 <- CSIdata.in4
for(i in 1:dim(CSIdata.in4)[1]){
  CSIdataREL2[i,] <- CSIdata.in4[i,]/sum(CSIdata.in4[i,])
}

dist.zoop <- vegdist(zoopREL, method = "bray")
dist.bact <- vegdist(CSIdataREL2, method = "bray")

mantel.rtest(dist.zoop, dist.bact, nrepet = 999)
```

```
## Warning in is.euclid(m1): Zero distance(s)
```

```
## Monte-Carlo test
## Observation: 0.4087451
## Call: mantelnoneuclid(m1 = m1, m2 = m2, nrepet = nrepet)
## Based on 999 replicates
## Simulated p-value: 0.001
```

```r
sampleREL.dist1 <- vegdist(CSIdataREL, method="bray")
# Principal Coordinates Analysis
CSI_pcoa1 <- cmdscale(sampleREL.dist1, k=3, eig=TRUE, add=FALSE)
  # Classical (Metric) Multidimensional Scaling; returns PCoA coordinates
  # eig=TRUE returns eigenvalues; k = # of dimensions to calculate

explainvar1a <- round(CSI_pcoa1$eig[1] / sum(CSI_pcoa1$eig), 3) * 100
explainvar2a <- round(CSI_pcoa1$eig[2] / sum(CSI_pcoa1$eig), 3) * 100
sum.eiga <- sum(explainvar1a, explainvar2a)

explainvar1a
```

```
## [1] 17.2
```

```r
explainvar2a
```

```
## [1] 7.4
```

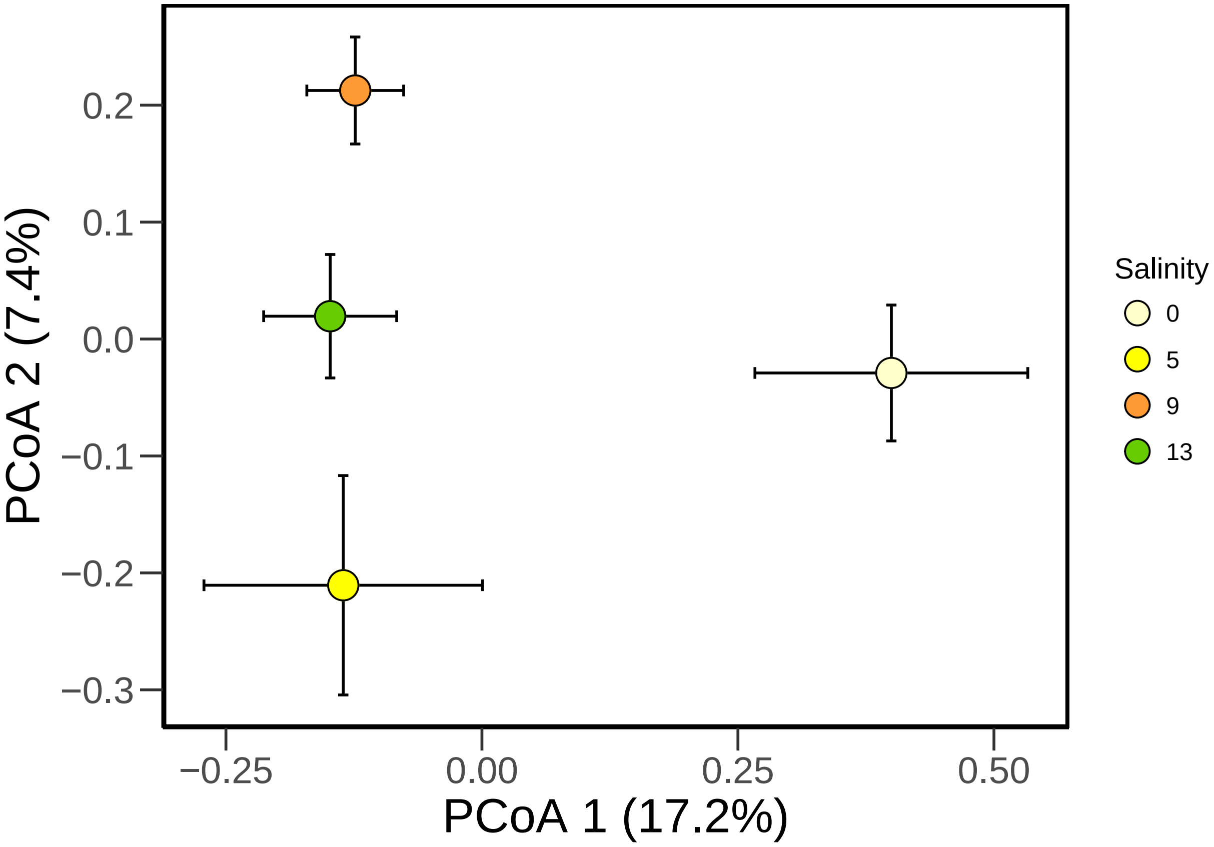
```r
#salinity
points1a <- cbind(as.data.frame(CSI_pcoa1$points), treatments1)
L.centroids1a <- melt(points1a, id="treatments1", measure.vars = c("V1", "V2"))
centroids1a <- cast(L.centroids1a, variable ~ treatments1, mean)
centroids.se1a <- cast(L.centroids1a, variable ~ treatments1, se)
centroids.sd1a <- cast(L.centroids1a, variable ~ treatments1, sd)

cent.dataframe1a <- t(data.frame(rbind(centroids1a[1,-1], centroids1a[2,-1],
                             centroids.sd1a[1,-1],centroids.sd1a[2,-1])))
colnames(cent.dataframe1a) <- c("V1", "V2", "V1e", "V2e")
cent.treats1a <- rownames(cent.dataframe1a)

##########################
#dispersal
points2 <- cbind(as.data.frame(CSI_pcoa1$points), treatments2)
L.centroids2 <- melt(points2, id="treatments2", measure.vars = c("V1", "V2"))
centroids2 <- cast(L.centroids2, variable ~ treatments2, mean)
centroids.se2 <- cast(L.centroids2, variable ~ treatments2, se)
centroids.sd2 <- cast(L.centroids2, variable ~ treatments2, sd)

cent.dataframe2 <- t(data.frame(rbind(centroids2[1,-1], centroids2[2,-1],
                             centroids.sd2[1,-1],centroids.sd2[2,-1])))
colnames(cent.dataframe2) <- c("V1", "V2", "V1e", "V2e")
cent.treats2 <- rownames(cent.dataframe2)
```

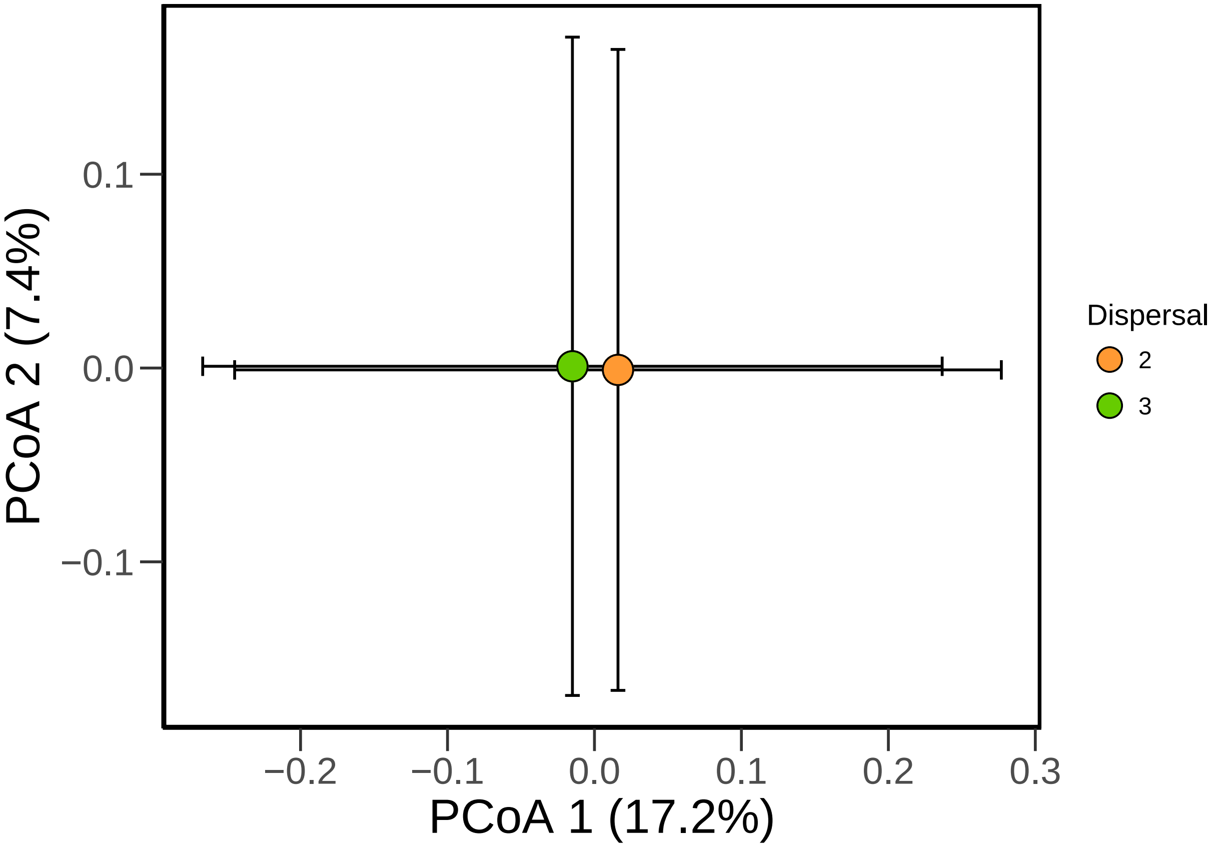
```
####################################
#salinity
df1a <- as.data.frame(cent.dataframe1a)
plot1a <- ggplot(df1a, aes(x=V1, y=V2, colour=cent.treats1a)) + theme_bw()
plot1a + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(), axis.line = elem
theme(panel.background = element_blank()) +
  geom_errorbarh(aes(xmax=V1+V1e, xmin=V1-V1e, height=0.01), colour="black") +
  geom_errorbar(aes(ymax=V2+V2e, ymin=V2-V2e, width=0.01), colour="black") +
  geom_point(size=5) +
  scale_colour_manual(labels = c("0","5","9","13"),   values = c("#FFFFCC", "#FFFF00", "#FF9933", "#66C
  geom_point(shape=1, size = 5,colour = "black") +
theme(axis.title=element_text(size=18), axis.text=element_text(size=14), axis.text.x    = element_text(s
  theme(axis.ticks.length=unit(0.3,"cm")) +
  xlab("PCoA 1 (17.2%)") + ylab("PCoA 2 (7.4%)") +
  labs(color = "Salinity") +
  guides(colour = guide_legend(override.aes = list(pch=21, size = 4, colour="black",
  fill=c("#FFFFCC", "#FFFF00", "#FF9933", "#66CC00"))))
```



```
ggsave("16SrRNA_CSI_Rplot_Salinity.pdf", plot=last_plot(), device=NULL, path=NULL, scale=1, width=NA, he
```

```
## Saving 6.5 x 4.5 in image
```

```
#####################################
#dispersal
df2 <- as.data.frame(cent.dataframe2)
plot2 <- ggplot(df2, aes(x=V1, y=V2, colour=cent.treats2)) + theme_bw()
plot2 + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(), axis.line = elemen
theme(panel.background = element_blank()) +
  geom_errorbarh(aes(xmax=V1+V1e, xmin=V1-V1e, height=0.01), colour="black") +
  geom_errorbar(aes(ymax=V2+V2e, ymin=V2-V2e, width=0.01), colour="black") +
  geom_point(size=5) +
  scale_colour_manual(labels = c("2","3"),   values = c("#FF9933", "#66CC00")) +
  geom_point(shape=1, size = 5,colour = "black") +
theme(axis.title=element_text(size=18), axis.text=element_text(size=14), axis.text.x = element_text(size
  theme(axis.ticks.length=unit(0.3,"cm")) +
  xlab("PCoA 1 (17.2%)") + ylab("PCoA 2 (7.4%)") +
  labs(color = "Dispersal") +
  guides(colour = guide_legend(override.aes = list(pch=21, size = 4, colour="black",
  fill=c("#FF9933", "#66CC00"))))
```



```
ggsave("16SrRNA_CSI_Rplot_Dispersal.pdf", plot=last_plot(), device=NULL, path=NULL, scale=1, width=NA, h
```

```
## Saving 6.5 x 4.5 in image
```

```
#How much bacterial variation is explained by salinity, N, P?
#bind design and bact files
```

```r
#newCSIdata <- cbind(design2,CSIdataREL) code from PERMANOVA section
# Log Transform Relative Abundances
df <- decostand(CSIdataREL, method="hellinger")
newCSIdata <- cbind(design2,CSIdataREL)

newCSIdata.2 <- na.omit(newCSIdata) #drop missing data NA for salinity
df.bcc <- newCSIdata.2[,-c(1:15)] #bacteria
df.env.bcc <- newCSIdata.2[,c(1:15)] #env
df.nuts <-newCSIdata.2[,c(9:11)] #nutrients

#To test significance salinity on bacterial community section 6.1 partition of variation based on redun

f <- df.bcc ~ Salinity_real + NH4um + NO3um + PO4um
df.rda <- rda(f, data=df.env.bcc)
anova(df.rda)
```

```
## Permutation test for rda under reduced model
## Permutation: free
## Number of permutations: 999
##
## Model: rda(formula = df.bcc ~ Salinity_real + NH4um + NO3um + PO4um, data = df.env.bcc)
##          Df Variance      F Pr(>F)
## Model     4 0.005942 2.1785  0.001 ***
## Residual 55 0.037503
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
mod <- varpart(df.bcc, ~ Salinity_real, df.nuts, data=df.env.bcc)
mod
```

```
##
## Partition of variance in RDA
##
## Call: varpart(Y = df.bcc, X = ~Salinity_real, df.nuts, data =
## df.env.bcc)
##
## Explanatory tables:
## X1:  ~Salinity_real
## X2:  df.nuts
##
## No. of explanatory tables: 2
## Total variation (SS): 2.5632
##             Variance: 0.043445
## No. of observations: 60
##
## Partition table:
##                      Df R.squared Adj.R.squared Testable
## [a+b] = X1            1   0.05645       0.04018     TRUE
## [b+c] = X2            3   0.10285       0.05479     TRUE
## [a+b+c] = X1+X2       4   0.13677       0.07399     TRUE
## Individual fractions
## [a] = X1|X2           1                 0.01920     TRUE
## [b]                   0                 0.02098    FALSE
## [c] = X2|X1           3                 0.03381     TRUE
## [d] = Residuals                         0.92601    FALSE
## ---
## Use function 'rda' to test significance of fractions of interest
```

```
#How much bacterial variation is explained by decomposition rates? - view into structure-function relat
#distance-based redundancy analysis bacterial community ~ decomposition rates for Date2=45 only
newCSIdata.3 <- subset(newCSIdata.2, Date2=="45")

df.bcc <- newCSIdata.3[,-c(1:15)] #bacteria
df.env.bcc <- newCSIdata.3[,c(1:15)] #env
df.decomp <- newCSIdata.3[,c(12:14)] #decomp

f <- df.bcc ~ Maple_dmass + Spartina_dmass + Phrag_dmass
df.rda <- rda(f, data=df.env.bcc)
anova(df.rda)
```

```
## Permutation test for rda under reduced model
## Permutation: free
## Number of permutations: 999
##
## Model: rda(formula = df.bcc ~ Maple_dmass + Spartina_dmass + Phrag_dmass, data = df.env.bcc)
##          Df Variance      F Pr(>F)
## Model     3 0.007498 1.5819  0.023 *
## Residual 26 0.041076
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#Is there a relationship betwen bacterial community composition and decomposition rate? used matrix com
dist.bcc <- vegdist(df.bcc, method = "bray")
dist.decomp <- vegdist(df.decomp, method = "euclidean")

mantel.rtest(dist.bcc, dist.decomp, nrepet = 999)
```

```
## Monte-Carlo test
## Observation: 0.2698445
## Call: mantel.rtest(m1 = dist.bcc, m2 = dist.decomp, nrepet = 999)
## Based on 999 replicates
## Simulated p-value: 0.001
```

```
########I did not update this diversity metrics section - AP
```