Resource Heterogeneity Structures Microbial Communities

Mario E. Muscarella

05 May, 2016

Introduction

What is this project about? What are they hypotheses?

Initial Setup

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

# Import Tools and Standard Functions
source("../bin/MothurTools.R")
source("../bin/CommonFunctions.R")

# Save Standard Plot Settings
opar <- par(no.readonly = TRUE)  # Saves plot defaults

# Load Required Packages
require("png"); require("grid");require("vegan"); require("igraph")
library("BiodiversityR", quietly = T);
require("picante");require("bioDist");require("gplots")
#require("xtable");require("png");require("grid");require("vegan")
#require("picante");require("phyloseq");require("car"); require("ade4")
#require("colorspace");require("bioDist");require("gplots")
#require("igraph");library("BiodiversityR")
```

Load Data & Minor Processing

Lake Nutrient Concentrations

```r
nuts <- read.csv(file = "../data/HMWF_Nutrients.txt", header = T)
```

Load DOM Profiles

```r
# Define Inputs
# Resource = raw site-by-resource matrix
resource.pos <- "../data/SpecAbundAvePos.csv"
```

```r
resource.neg <- "../data/SpecAbundAveNeg.csv"
design.in <- "../data/design.txt"

# Import Design
design <- read.delim(design.in, header=T, row.names=1)

# Import Resources
res.in <- read.csv(resource.neg, header=T, row.names=1)
rownames(res.in) <- c("Ann", "blank", "CanyonChemo", "Canyon", "CanyonHypo",
                      "CanyonI", "CanyonII", "CanyonIII", "CanyonIV", "Howe",
                      "Ives", "Jordan", "Lily", "Mountain", "Pony", "Rush",
                      "SecondPine", "UpperPine")

blank <- unlist(res.in["blank", ])
res.hmwf <- res.in[-c(which(rownames(res.in) %in% c("blank", "CanyonChemo",
                      "CanyonHypo", "CanyonI", "CanyonII",
                      "CanyonIII", "CanyonIV", "Jordan"))), ]

# Remove Blank Peaks
for (i in 1:dim(res.hmwf)[1]){
  res.hmwf[i, ] <- res.hmwf[i, ] - blank * 1.1
}

# Remove Peaks Under Height of 50
res.hmwf[res.hmwf < 50] <- 0

# Remove Zero Sum Columns
res.hmwf <- res.hmwf[,colSums(res.hmwf) > 0]

# Data Transformations
# Reorder Sites
res <- res.hmwf[order(rownames(res.hmwf)), ]

# Sequencing Coverage
coverage <- rowSums(res)
resources <- dim(res)[2]

# Make Relative Abundence Matrices
resREL <- res
for(i in 1:dim(res)[1]){
  resREL[i,] <- res[i,]/sum(res[i,])
}

# Log Transform Relative Resource Abundance
resREL.log <- decostand(resREL, method="log")
```

```
## Warning: non-integer data: divided by smallest positive value
```

Load Bacterial Community Data

```r
# Define Inputs
# Design = general design file for experiment
# shared = OTU table from mothur with sequence similarity clustering
# Taxonomy = Taxonomic information for each OTU
design.in <- "../data/design.txt"
shared    <- "../data/HMWF.bac.final.shared"
taxon     <- "../data/HMWF.bac.final.0.03.taxonomy"

# Import Design
design <- read.delim(design.in, header=T, row.names=1)

# Import Shared Files
OTUs.in <- read.otu(shared = shared, cutoff = "0.03")     # 97% Similarity

# Import Taxonomy
OTU.tax <- read.tax(taxonomy = taxon, format = "rdp")

# Data Transformations
# Reorder Site
OTUs.hmwf <- OTUs.in[rownames(design), ]

# Remove OTUs with less than two occurences across all sites
# OTUs <- OTUs.hmwf[, which(colSums(OTUs.hmwf) >= 2)]
OTUs <- OTUs.hmwf[, colSums((OTUs.hmwf > 0) * 1)  >= 2 | colSums(OTUs.hmwf) >= 10]

# Sequencing Coverage
coverage <- rowSums(OTUs)
bacteria <- dim(OTUs)[2]

# Good's Coverage
goods.c <- goods(OTUs)

# Make Presence Absence Matrix
OTUsPA <- (OTUs > 0) * 1

# Make Relative Abundence Matrices
OTUsREL <- OTUs
for(i in 1:dim(OTUs)[1]){
  OTUsREL[i,] <- OTUs[i,]/sum(OTUs[i,])
}

# Log Transform Relative Abundances
OTUsREL.log <- decostand(OTUs, method="log")
```

Statistical Description of Resources

```r
range(nuts$DOC);range(nuts$TN);range(nuts$TP)
```

```
## [1]  4.22 30.46
```

```
## [1] 0.30 1.86
```

```
## [1]  1.35 17.04
```

```
CV(nuts$DOC)
```

```
## [1] 75.80279
```

```
CV(nuts$TN)
```

```
## [1] 67.33346
```

```
CV(nuts$TP)
```

```
## [1] 75.73826
```

```
cor.test(nuts$DOC, nuts$TN)
```

```
##
##  Pearson's product-moment correlation
##
## data:  nuts$DOC and nuts$TN
## t = 17.683, df = 18, p-value = 7.975e-13
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9301241 0.9892442
## sample estimates:
##       cor
## 0.9724041
```

```
cor.test(nuts$TN, nuts$TP)
```

```
##
##  Pearson's product-moment correlation
##
## data:  nuts$TN and nuts$TP
## t = 4.4208, df = 18, p-value = 0.00033
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4098173 0.8823129
## sample estimates:
##       cor
## 0.7214933
```

```
cor.test(nuts$DOC, nuts$TP)
```

```
##
##  Pearson's product-moment correlation
##
## data:  nuts$DOC and nuts$TP
```

```
## t = 3.9123, df = 18, p-value = 0.001021
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3362582 0.8618749
## sample estimates:
##       cor
## 0.6779058
```

```r
# Principal Components Axis
nuts.pca <- princomp(nuts[, 4:5])
summary(nuts.pca)
```

```
## Importance of components:
##                          Comp.1      Comp.2
## Standard deviation     4.5605793 0.276415472
## Proportion of Variance 0.9963399 0.003660094
## Cumulative Proportion  0.9963399 1.000000000
```

```r
nuts.axis <- nuts.pca$scores[,1]

PCA.res <- princomp(cbind(scale(nuts$DOC), scale(nuts$TN), scale(nuts$TP)))
summary(PCA.res)
```

```
## Importance of components:
##                          Comp.1    Comp.2      Comp.3
## Standard deviation     1.5682426 0.6052381 0.155890961
## Proportion of Variance 0.8629421 0.1285309 0.008527015
## Cumulative Proportion  0.8629421 0.9914730 1.000000000
```

```r
PCA.res1 <- scale(PCA.res$scores[,1])
```

Statistical Description of DOM Structural Diversity

```r
# Observed Richness
S.res <- rowSums((res > 0) * 1)

# Simpson's Evenness
res.simpsE <- round(apply(res, 1, SimpE), 3)

# Shannon's Diversity
res.shan <- round(vegan::diversity(res, index = "shannon"), 2)

# Combine Alpha Diversity
res.div <- as.data.frame(cbind(S.res, res.simpsE, res.shan))

# Summary Stats
range(res.div$S.res);range(res.div$res.shan);range(res.div$res.simpsE)
```

```
## [1] 529 569
```

```
## [1] 4.89 5.56
```

```
## [1] 0.053 0.152
```

```
CV(res.div$S.res);CV(res.div$res.shan);CV(res.div$res.simpsE)
```

```
## [1] 2.118764
```

```
## [1] 3.629068
```

```
## [1] 32.91581
```

DOM Compositional Diversity

```r
# Calculate Bray-Curtis
hmwf.bray.res <- vegdist(resREL, method = "bray")
# hmwf.bray.res <- vegdist(resREL.log, method = "bray")

dis.mean <- mean(hmwf.bray.res)

# Principal Coordinates Analysis
pcoa.res <- cmdscale(hmwf.bray.res, eig = TRUE, k = 3)
explainvar1.res <- round(pcoa.res$eig[1] / sum(pcoa.res$eig), 3) * 100
explainvar2.res <- round(pcoa.res$eig[2] / sum(pcoa.res$eig), 3) * 100
explainvar3.res <- round(pcoa.res$eig[3] / sum(pcoa.res$eig), 3) * 100
sum.eig.res <- sum(explainvar1.res, explainvar2.res, explainvar3.res)

# DOM Scores
dom.scores <- add.spec.scores(pcoa.res,res,method="cor.scores",multi=1,Rscale=F,scaling="1")
dom.scores <- as.matrix(dom.scores$cproj)[,1:2]
dom.scores <- dom.scores[abs(dom.scores[,1]) > 0.7 | abs(dom.scores[,2]) > 0.7, ]

write.table(round(dom.scores, 3), file = "../data/HMWF_DOM.txt", sep = "\t", quote = F,
            col.names = NA)
```

Statistical Description of Bacterial Structural Diversity

```r
# Observed Richness
S.obs <- rowSums((OTUs > 0) * 1)

# Simpson's Evenness
simpsE <- round(apply(OTUs, 1, SimpE), 3)

# Shannon's Diversity
shan <- vegan::diversity(OTUs, index = "shannon")

# Rarefied Richness
```

```
S.rar <- round(rarefy(OTUs, min(rowSums(OTUs))), 0)

alpha.div <- cbind(design, S.obs, simpsE, shan, S.rar)
alpha.div <- alpha.div[order(alpha.div$Lake, alpha.div$Year, alpha.div$Molecule), ]

# Summary Stats
range(alpha.div$S.rar);range(alpha.div$simpsE)
```

```
## [1]  521 1956
```

```
## [1] 0.004 0.054
```

```
CV(alpha.div$S.rar);CV(alpha.div$simpsE)
```

```
## [1] 39.57749
```

```
## [1] 55.37234
```

```
CV(alpha.div$S.rar[alpha.div$Lake != "Pony" & alpha.div$Lake != "Lily"])
```

```
## [1] 16.67472
```

```
range(alpha.div$S.rar[alpha.div$Lake != "Pony" & alpha.div$Lake != "Lily"])
```

```
## [1] 521 907
```

Bacterial Compositional Diversity

```
# Calculate Bray-Curtis
hmwf.bray.REL <- vegdist(OTUsREL, method = "bray")

dis.mean <- mean(hmwf.bray.REL)

adonis(hmwf.bray.REL ~ design$Molecule)
```

```
##
## Call:
## adonis(formula = hmwf.bray.REL ~ design$Molecule)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##                 Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
## design$Molecule  1    0.8096 0.80965  4.6191 0.10838  0.001 ***
## Residuals       38    6.6608 0.17528         0.89162
## Total           39    7.4705                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Principal Coordinates Analysis
pcoa.rel <- cmdscale(hmwf.bray.REL, eig = TRUE, k = 3)
explainvar1.rel <- round(pcoa.rel$eig[1] / sum(pcoa.rel$eig), 3) * 100
explainvar2.rel <- round(pcoa.rel$eig[2] / sum(pcoa.rel$eig), 3) * 100
explainvar3.rel <- round(pcoa.rel$eig[3] / sum(pcoa.rel$eig), 3) * 100
sum.eig.rel <- sum(explainvar1.rel, explainvar2.rel, explainvar3.rel)


# OTU Scores
otu.scores <- add.spec.scores(pcoa.rel,OTUsREL,method="cor.scores",multi=1,Rscale=F,scaling="1")
otu.scores <- as.matrix(otu.scores$cproj)[,1:2]
otu.scores <- otu.scores[abs(otu.scores[,1]) > 0.65 | abs(otu.scores[,2]) > 0.65, ]
```

Resource Heterogeneity and Community Diversity

Structural Relationships

```
# Organize Data
nuts$PCA <- as.numeric(PCA.res1 + 1)
dat1D <- data.frame(alpha.div[alpha.div$Molecule == "DNA", ], nuts[order(nuts$Site), ])
dat1R <- data.frame(alpha.div[alpha.div$Molecule == "RNA", ], nuts[order(nuts$Site), ])
dat2D <- data.frame(dat1D[dat1D$Year == "2012", ], res.div[order(rownames(res.div)), ])
dat2R <- data.frame(dat1R[dat1R$Year == "2012", ], res.div[order(rownames(res.div)), ])
dat2  <- data.frame(rbind(dat2D, dat2R))
dat2  <- dat2[order(dat2$Lake), ]

# Resource Concentration and Diversity (Total)
mod1 <- lm(S.rar ~ PCA, data = dat1D[dat1D$Lake != "Pony", ])
mod2 <- lm(simpsE ~ PCA, data = dat1D[dat1D$Lake != "Pony", ])
summary(mod1);summary(mod2)
```

```
##
## Call:
## lm(formula = S.rar ~ PCA, data = dat1D[dat1D$Lake != "Pony",
##     ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -202.10  -82.01   35.64   60.98  273.67
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   626.20      58.40  10.722 1.03e-08 ***
## PCA            92.22      70.78   1.303    0.211
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 128.4 on 16 degrees of freedom
## Multiple R-squared:  0.09592,    Adjusted R-squared:  0.03942
## F-statistic: 1.698 on 1 and 16 DF,  p-value: 0.211
```

```
##
## Call:
## lm(formula = simpsE ~ PCA, data = dat1D[dat1D$Lake != "Pony",
##      ])
##
## Residuals:
##        Min        1Q    Median        3Q       Max
## -0.008897 -0.007358 -0.005585  0.004540  0.031805
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.024029   0.005106   4.706 0.000238 ***
## PCA         -0.004844   0.006189  -0.783 0.445223
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01123 on 16 degrees of freedom
## Multiple R-squared:  0.03688,    Adjusted R-squared:  -0.02332
## F-statistic: 0.6127 on 1 and 16 DF,  p-value: 0.4452
```

```r
# Resource Heterogeneity and Divesity (Total)
mod3 <- lm(S.rar ~ S.res, data = dat2D[dat2D$Lake != "Pony", ])
mod4 <- lm(simpsE ~ S.res, data = dat2D[dat2D$Lake != "Pony", ])
summary(mod3);summary(mod4)
```

```
##
## Call:
## lm(formula = S.rar ~ S.res, data = dat2D[dat2D$Lake != "Pony",
##      ])
##
## Residuals:
##      Min      1Q  Median      3Q     Max
## -181.63 -105.75   22.00   35.25  329.63
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6727.625   3953.770   1.702    0.133
## S.res        -10.875      7.089  -1.534    0.169
##
## Residual standard error: 159.2 on 7 degrees of freedom
## Multiple R-squared:  0.2516, Adjusted R-squared:  0.1447
## F-statistic: 2.353 on 1 and 7 DF,  p-value: 0.1689
```

```
##
## Call:
## lm(formula = simpsE ~ S.res, data = dat2D[dat2D$Lake != "Pony",
##      ])
##
## Residuals:
##         Min         1Q     Median         3Q        Max
## -0.0094691 -0.0072840 -0.0006914  0.0064568  0.0122716
##
## Coefficients:
```

9

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.7590864  0.2110552  -3.597  0.00878 **
## S.res        0.0014074  0.0003784   3.719  0.00746 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.008496 on 7 degrees of freedom
## Multiple R-squared:  0.664,  Adjusted R-squared:  0.616
## F-statistic: 13.83 on 1 and 7 DF,  p-value: 0.007465
```

```r
# Same Tests with Active Community
mod5 <- lm(S.rar ~ PCA, data = dat1R[dat1R$Lake != "Pony", ])
mod6 <- lm(simpsE ~ PCA, data = dat1R[dat1R$Lake != "Pony", ])
mod7 <- lm(S.rar ~ S.res, data = dat2R[dat2R$Lake != "Pony", ])
mod8 <- lm(simpsE ~ S.res, data = dat2R[dat2R$Lake != "Pony", ])
summary(mod8)
```

```
##
## Call:
## lm(formula = simpsE ~ S.res, data = dat2R[dat2R$Lake != "Pony",
##     ])
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.012859 -0.006603  0.001504  0.007206  0.008034
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.6079861  0.2037445  -2.984   0.0204 *
## S.res        0.0011488  0.0003653   3.145   0.0163 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.008201 on 7 degrees of freedom
## Multiple R-squared:  0.5855, Adjusted R-squared:  0.5263
## F-statistic: 9.889 on 1 and 7 DF,  p-value: 0.01627
```

```r
# Stats
mod4.p <- round(summary(mod4)$coefficients[2,4], 3)

# Prediction Frames
pred.frame1 <- data.frame(PCA = seq(0, 2.1, 0.1))
pred.frame2 <- data.frame(S.res = seq(542,572,2))

# Correlation Test
cor.test(~ S.res + PCA, data = dat2D[dat2D$Lake != "Pony", ])
```

```
##
##  Pearson's product-moment correlation
##
## data:  S.res and PCA
## t = -0.67966, df = 7, p-value = 0.5186
## alternative hypothesis: true correlation is not equal to 0
```

```
## 95 percent confidence interval:
##  -0.7834698  0.4975245
## sample estimates:
##        cor
## -0.2488074
```

Structural Relationship Plots

```r
# Confidence Hulls
add.hull <- function(model = "", pred.frame = ""){
  CI.U <- predict(model, interval = "c", newdata=pred.frame)[, "upr"]
  CI.L <- predict(model, interval = "c", newdata=pred.frame)[, "lwr"]
  pred.frame2 <- unlist(pred.frame)
  X.Vec <- c(pred.frame2, tail(pred.frame2, 1), rev(pred.frame2),
             head(pred.frame2, 1))
  Y.Vec <- c(CI.U, tail(CI.L, 1), rev(CI.L), head(CI.U,1))
  polygon(X.Vec, Y.Vec, col = "gray90", border = NA)
}

png(filename="../figures/Figure1.png",
    width = 1600, height = 1600, res = 96*2, bg = "white")
par(opar)

layout(matrix(1:4, nrow = 2, byrow = F))
par(mar = c(0.5, 1, 1, 1) + 0.1, oma = c(5, 5.5, 0, 0) + 0.1)

# Resource Concentration vs Species Richness
plot(dat1D$S.rar ~ dat1D$PCA,
     xlab = "", ylab = "", axes = F,
     xlim = c(0, 2.1), ylim = c(400, 1200), las = 1,
     pch = 22, col = "black", bg = "gray", cex = 2, lwd = 2)
add.hull(model = mod1, pred.frame = pred.frame1)
matlines(pred.frame1, predict(mod1, interval = "c", newdata=pred.frame1),
         lty=c(2,3,3), lwd=c(4,2,2), col="black")
points(dat1D$S.rar ~ dat1D$PCA,
       pch = 22, col = "black", bg = "gray", cex = 2, lwd = 2)
legend("bottomright", legend = bquote(italic(N.S.)),
       bty = "n", cex = 1, inset = 0.01)
# mtext("Nutrients", side = 1, line = 3, cex = 1.5)
mtext("Species Richness", side = 2, line = 4, cex = 1.5)
axis(1, lwd = 2, labels = F, at = c(0, 2))
axis(1, lwd = 2, tck = -0.02, labels = F)
axis(2, lwd = 2, labels = T, las = 1)
axis(3, lwd = 2, tck = -0.02, labels = F)
axis(4, lwd = 2, tck = -0.02, labels = F)
axis(1, lwd = 2, tck = 0.02, labels = F)
axis(2, lwd = 2, tck = 0.02, labels = F)
axis(3, lwd = 2, tck = 0.02, labels = F)
axis(4, lwd = 2, tck = 0.02, labels = F)
box(lwd = 2)

# Resource Concentration vs Species Evenness
```

```r
plot(dat1D$simpsE ~ dat1D$PCA,
     xlab = "", ylab = "", axes = F,
     xlim = c(0, 2.1), ylim = c(0, 0.06), las = 1,
     pch = 22, col = "black", bg = "gray", cex = 2, lwd = 2)
add.hull(model = mod2, pred.frame = pred.frame1)
matlines(pred.frame1, predict(mod2, interval = "c", newdata=pred.frame1),
         lty=c(2,3,3), lwd=c(4,2,2), col="black")
points(dat1D$simpsE ~ dat1D$PCA,
       pch = 22, col = "black", bg = "gray", cex = 2, lwd = 2)
legend("topright", legend = bquote(italic(N.S.)),
       bty = "n", cex = 1, inset = 0.01)
mtext("Resource Concentration", side = 1, line = 3.5, cex = 1.5)
mtext("Species Evenness", side = 2, line = 4, cex = 1.5)
axis(1, lwd = 2, labels = c("low", "high"), at = c(0, 2))
axis(1, lwd = 2, tck = -0.02, labels = F)
axis(2, lwd = 2, labels = T, las = 1, at = c(seq(0, 0.06, 0.02)))
axis(3, lwd = 2, tck = -0.02, labels = F)
axis(4, lwd = 2, tck = -0.02, labels = F, at = c(seq(0, 0.06, 0.02)))
axis(1, lwd = 2, tck = 0.02, labels = F)
axis(2, lwd = 2, tck = 0.02, labels = F, at = c(seq(0, 0.06, 0.02)))
axis(3, lwd = 2, tck = 0.02, labels = F)
axis(4, lwd = 2, tck = 0.02, labels = F, at = c(seq(0, 0.06, 0.02)))
box(lwd = 2)

# Resource Richness vs Species Richness
plot(dat2D$S.rar ~ dat2D$S.res,
     xlab = "", ylab = "", type = "n", axes = F,
     xlim = c(540, 572), ylim = c(400, 1200), las = 1,
     pch = 22, col = "black", bg = "gray", cex = 2, lwd = 2)
add.hull(model = mod3, pred.frame = pred.frame2)
matlines(pred.frame2, predict(mod3, interval = "c", newdata=pred.frame2),
         lty=c(2,3,3), lwd=c(4,2,2), col="black")
points(dat2D$S.rar ~ dat2D$S.res,
       pch = 22, col = "black", bg = "gray", cex = 2, lwd = 2)
legend("topright", legend = bquote(italic(N.S.)),
       bty = "n", cex = 1, inset = 0.01)
# mtext("DOM Richness", side = 1, line = 3, cex = 1.5)
# mtext("Species Richness", side = 2, line = 3.5, cex = 1.5)
axis(1, lwd = 2, labels = F, at = c(seq(540, 570, 10)))
axis(2, lwd = 2, labels = F, las = 1)
axis(3, lwd = 2, tck = -0.02, labels = F, at = c(seq(540, 570, 10)))
axis(4, lwd = 2, tck = -0.02, labels = F)
axis(1, lwd = 2, tck = 0.02, labels = F, at = c(seq(540, 570, 10)))
axis(2, lwd = 2, tck = 0.02, labels = F)
axis(3, lwd = 2, tck = 0.02, labels = F, at = c(seq(540, 570, 10)))
axis(4, lwd = 2, tck = 0.02, labels = F)
box(lwd = 2)

# Resource Richness vs Species Eveness
plot(dat2D$simpsE ~ dat2D$S.res,
     xlab = "", ylab = "", type = "n", axes = F,
     xlim = c(540, 572), ylim = c(0, 0.06), las = 1,
     pch = 22, col = "black", bg = "gray", cex = 2, lwd = 2)
```

```
add.hull(model = mod4, pred.frame = pred.frame2)
matlines(pred.frame2, predict(mod4, interval = "c", newdata=pred.frame2),
         lty=c(2,3,3), lwd=c(4,2,2), col="black")
points(dat2D$simpsE ~ dat2D$S.res,
       pch = 22, col = "black", bg = "gray", cex = 2, lwd = 2)
legend("bottomright", legend = bquote(italic(P) == .(mod4.p)),
       bty = "n", cex = 1, inset = 0.01)
mtext("DOM Richness", side = 1, line = 3.5, cex = 1.5)
# mtext("Species Evenness", side = 2, line = 3.5, cex = 1.5)
axis(1, lwd = 2, labels = T, las = 1, at = c(seq(540, 570, 10)))
axis(2, lwd = 2, labels = F, las = 1, at = c(seq(0, 0.06, 0.02)))
axis(3, lwd = 2, tck = -0.02, labels = F, at = c(seq(540, 570, 10)))
axis(4, lwd = 2, tck = -0.02, labels = F, at = c(seq(0, 0.06, 0.02)))
axis(1, lwd = 2, tck = 0.02, labels = F, at = c(seq(540, 570, 10)))
axis(2, lwd = 2, tck = 0.02, labels = F, at = c(seq(0, 0.06, 0.02)))
axis(3, lwd = 2, tck = 0.02, labels = F, at = c(seq(540, 570, 10)))
axis(4, lwd = 2, tck = 0.02, labels = F, at = c(seq(0, 0.06, 0.02)))
box(lwd = 2)

dev.off() # this writes plot to folder
graphics.off() # shuts down open devices
par(opar)
```
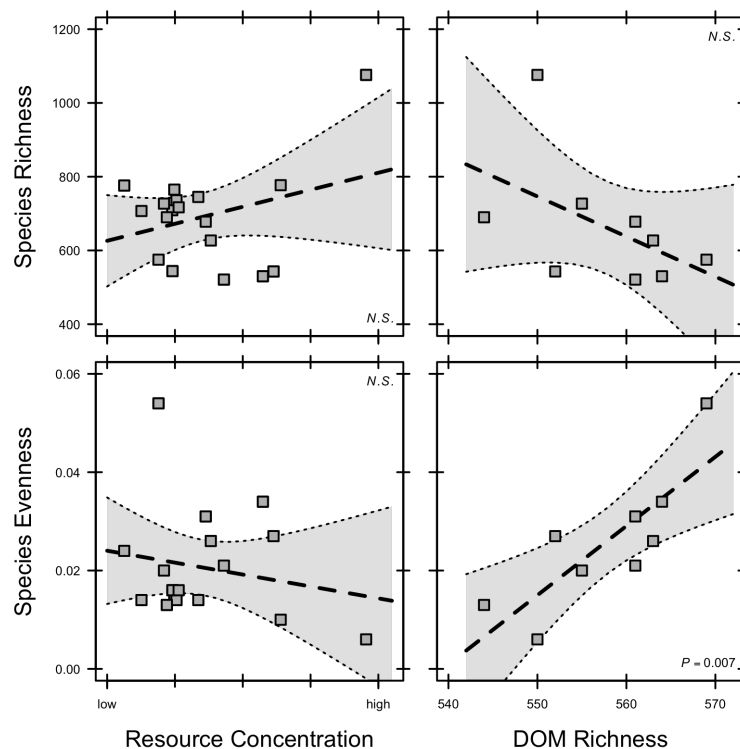

Figure 1: Diversity

Compositional Relationships (dbRDA)

```r
# Define Environmental Gradients
## Resource Concentration
## dat1D; dat1R; dat2D; dat2R

## DOM Composition
## pcoa.res$points[, 1]; pcoa.res$points[, 2]; pcoa.res$points[, 3]

## Community Composition
# Define DNA and RNA Community
OTUsREL.D <- OTUsREL[design$Molecule == "DNA" , ]
OTUsREL.R <- OTUsREL[design$Molecule == "RNA" , ]
OTUsREL.D2012 <- OTUsREL[design$Molecule == "DNA" & design$Year == "2012" , ]
OTUsREL.R2012 <- OTUsREL[design$Molecule == "RNA" & design$Year == "2012" , ]

# Calculate Bray-Curtis Distances for Bacteria
Bray.REL.D <- vegdist(decostand(OTUsREL.D, "log"), "bray")
Bray.REL.R <- vegdist(decostand(OTUsREL.R, "log"), "bray")
Bray.REL.D2012 <- vegdist(decostand(OTUsREL.D2012, "log"), "bray")
Bray.REL.R2012 <- vegdist(decostand(OTUsREL.R2012, "log"), "bray")

# Resource Concentration dbRDA
dbRDA.D <- capscale(Bray.REL.D ~ dat1D$PCA, comm = OTUsREL.D, add = T)
dbRDA.R <- capscale(Bray.REL.R ~ dat1R$PCA, comm = OTUsREL.R, add = T)
bdRDA.D2012 <- capscale(Bray.REL.D2012 ~ dat2D$PCA, comm = OTUsREL.D2012, add = T)
dbRDA.R2012 <- capscale(Bray.REL.R2012 ~ dat2R$PCA, comm = OTUsREL.R2012, add = T)

anova(bdRDA.D2012, permutations = how(nperm=9999))
RsquareAdj(bdRDA.D2012)
anova(dbRDA.R2012, permutations = how(nperm=9999))
RsquareAdj(dbRDA.R2012)

# With Pony and Lily Removed (DNA)
dat2D.2 <- dat2D[dat2D$Lake != "Pony" & dat2D$Lake != "Lily", ]
OTUsREL.D.2 <- OTUsREL[design$Molecule == "DNA" & design$Year == "2012" &
                       design$Lake != "Pony" & design$Lake != "Lily", ]
Bray.REL.D.2 <- vegdist(OTUsREL.D.2, "bray")
dbRDA.D.2 <- capscale(Bray.REL.D.2 ~ dat2D.2$PCA, comm = OTUsREL.D.2, add = T)

anova(dbRDA.D.2, permutations = how(nperm=9999))
RsquareAdj(dbRDA.D.2)

# DOM Diversity dbRDA
Bray.RES <- vegdist(decostand(resREL, "log"), "bray")
pcoa.RES <- cmdscale(Bray.RES, k = 3, eig = T)
dbRDA.D.DOM <- capscale(Bray.REL.D2012 ~ pcoa.RES$points,
                        comm = OTUsREL.D2012, add = T)
dbRDA.R.DOM <- capscale(Bray.REL.R2012 ~ pcoa.RES$points,
                        comm = OTUsREL.R2012, add = T)

anova(dbRDA.D.DOM, permutations = how(nperm=9999))
RsquareAdj(dbRDA.D.DOM)
```

```r
anova(dbRDA.R.DOM, permutations = how(nperm=9999))
RsquareAdj(dbRDA.R.DOM)

# With Pony and Lily Removed (DNA)
## Subset OTUs
OTUsREL.D.2 <- OTUsREL[design$Molecule == "DNA" & design$Year == "2012" &
                        design$Lake != "Pony" & design$Lake != "Lily", ]
rownames(OTUsREL.D.2) <- gsub("2012_DNA", "", rownames(OTUsREL.D.2))
Bray.REL.D2012.2 <- vegdist(decostand(OTUsREL.D.2, "log"), method = "bray")

## Subset Resources
resREL2 <- resREL[rownames(resREL) != "Pony" & rownames(resREL) != "Lily", ]
Bray.RES2 <- vegdist(resREL2, method = "bray")
pcoa.RES2 <- cmdscale(Bray.RES2, eig = TRUE, k = 2)

## dbRDA Analysis
dbRDA.D.DOM.2 <- capscale(Bray.REL.D2012.2 ~ pcoa.RES2$points, add = T)
anova(dbRDA.D.DOM.2, permutations = how(nperm=9999), model = "direct")
RsquareAdj(dbRDA.D.DOM.2)

# Resource Concentration/Composition Correlations
cor.test(dat2D$PCA, pcoa.res$points[,1])
cor.test(dat2D$PCA, pcoa.res$points[,2])
cor.test(dat2D$PCA, pcoa.res$points[,3])
```

Bacterial PCoA Plot with Env Vectors

```r
# PCoA of Total Community
bray.BAC <- vegdist(decostand(OTUsREL, "log"), "bray")
```

```
## Warning: non-integer data: divided by smallest positive value
```

```r
pcoa.BAC <- cmdscale(Bray.REL.D2012, k = 2, eig = T)
explainvar1 <- round(pcoa.BAC$eig[1] / sum(pcoa.BAC$eig), 3) * 100
explainvar2 <- round(pcoa.BAC$eig[2] / sum(pcoa.BAC$eig), 3) * 100

# PCoA of Resoruces
bray.RES <- vegdist(decostand(resREL, "log"), "bray")
```

```
## Warning: non-integer data: divided by smallest positive value
```

```r
pcoa.RES <- cmdscale(bray.RES, k = 3, eig = T)

# Resource Concentrations
cons.RES <- dat2D$PCA

# Initial Plot as PNG
png(filename="../figures/Figure2.png",
    width = 1300, height = 900, res = 96*2, bg = "white")
```

```r
# Define Plot Parameters
par(opar)
par(mar = c(4, 5, 1, 1) + 0.5)

# Initiate Plot 1
plot(pcoa.BAC$points[ ,1], pcoa.BAC$points[ ,2],
     ylim = c(-0.4, 0.4), xlim = c(-0.4, 0.6),
     xlab = paste("PCoA 1 (", explainvar1, "%)", sep = ""),
     ylab = paste("PCoA 2 (", explainvar2, "%)", sep = ""),
     #xlab = "", ylab = "", xaxt = "n", yaxt = "n",
     pch = 22, cex = 2.0, type = "n", cex.lab = 1.5, cex.axis = 1,
     axes = FALSE)

# Add Axes
axis(side = 1, labels = T, lwd.ticks = 2, cex.axis = 1, las = 1)
axis(side = 2, labels = T, lwd.ticks = 2, cex.axis = 1, las = 1)
axis(side = 3, labels = F, lwd.ticks = 2, cex.axis = 1, las = 1, tck=-0.02)
axis(side = 4, labels = F, lwd.ticks = 2, cex.axis = 1, las = 1, tck=-0.02)
axis(side = 1, labels = F, lwd.ticks = 2, cex.axis = 1, las = 1, tck=0.01)
axis(side = 2, labels = F, lwd.ticks = 2, cex.axis = 1, las = 1, tck=0.01)
axis(side = 3, labels = F, lwd.ticks = 2, cex.axis = 1, las = 1, tck=0.01)
axis(side = 4, labels = F, lwd.ticks = 2, cex.axis = 1, las = 1, tck=0.01)
abline(h = 0, v = 0, lty = 3)
box(lwd = 2)

# Add Points & Labels
points(pcoa.BAC$points[ ,1], pcoa.BAC$points[ ,2], pch = 22,
       cex = 2, bg = "gray", lwd = 2)

# Resource Concentration Vector
res.con <- envfit(pcoa.BAC, cons.RES)
con.arrows <- res.con[[1]]$arrows * 0.5
arrows(0, 0, con.arrows[, 1], con.arrows[, 2], col = "red", length = 0.1, lwd = 2)
text(con.arrows[, 1] * 1.2, con.arrows[, 2] * 1.2, "Conc.", col = "red", cex = 1)

# DOM Composition Vectors
res.com <- envfit(pcoa.BAC, pcoa.RES$points[,1:2])
com.arrows <- res.com[[1]]$arrows * 0.3
arrows(0, 0, com.arrows[1, 1], com.arrows[1, 2], col = "red", length = 0.1, lwd = 2)
arrows(0, 0, com.arrows[2, 1], com.arrows[2, 2], col = "red", length = 0.1, lwd = 2)
text(com.arrows[1, 1] * 1.15, com.arrows[1, 2] * 1.15, "DOM 1", col = "red", cex = 1)
text(com.arrows[2, 1] * 1, com.arrows[2, 2] * 1.15, "DOM 2", col = "red",
     cex = 1, pos = 1)

text(0.45, -0.1, "Pony", col = "black", cex = 0.8)
text(0.55, 0.15, "Lily", col = "black", cex = 0.8)

par(opar)
dev.off() # this writes plot to folder
graphics.off() # shuts down open devices
```
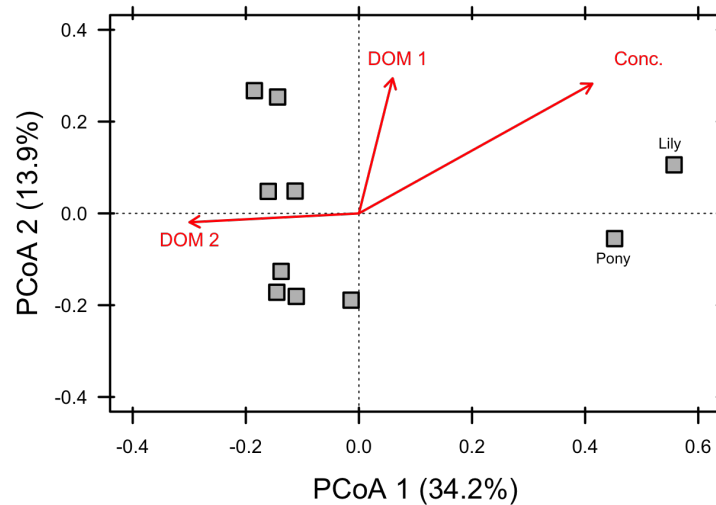
Figure 2: Diversity

Consumer Resource Specialization

```
# Subset OTUs for most dominant
OTUsREL.dom <- OTUsREL[,which(colSums(as.matrix(OTUsREL)) > 0.05)]
OTUsREL.dom2012 <- OTUsREL.dom[grep("2012_RNA", rownames(OTUsREL.dom)),]
OTUsREL.dom2012 <- OTUsREL.dom2012[,order(colSums(as.matrix(OTUsREL.dom2012)),
                                   decreasing = T)[1:100]]

# Subset Resources for most dominant
resREL.dom <- resREL[,which(colSums(as.matrix(resREL)) > 0.022)]
resREL.dom <- resREL.dom[,order(colSums(as.matrix(resREL.dom)), decreasing = T)[1:100]]

# Subset Resoruce for most influential
resREL.inf <- add.spec.scores(pcoa.RES,resREL,method="cor.scores",multi=1,Rscale=F,scaling="1")
resREL.inf <- as.matrix(resREL.inf$cproj)[,1:3]
resREL.inf <- resREL.inf[abs(resREL.inf[,1]) > 0.7 | abs(resREL.inf[,2]) > 0.7 |
                         abs(resREL.inf[,3]) > 0.7, ]

# Mege OTUs and Resources Matrices
ConRes1 <- cbind(as.matrix(OTUsREL.dom2012), as.matrix(resREL.dom))
ConRes2 <- cbind(as.matrix(resREL.dom), as.matrix(OTUsREL.dom2012))
ConRes <- rbind(ConRes1, ConRes2)

# Calculate 1 - Spearman Correlation Coefficients: Spearman Distance
spear.ConRes <- spearman.dist(t(as.matrix(ConRes)), abs = FALSE, diag = T, upper = T)

# Isolate Relavent Interactions
spear.ConRes2 <- spear.ConRes - 1
spear.ConRes3 <- spear.ConRes2
spear.ConRes3[which(spear.ConRes2 < 0.5 & spear.ConRes2 > -0.5)] <- 0
spear.ConRes4 <- as.matrix(spear.ConRes3)[1:100, 101:200]
```

```
## Warning in as.matrix.dist(spear.ConRes3): number of items to replace is not
## a multiple of replacement length
```

```r
# Summary
str(spear.ConRes4)
```

```
##  num [1:100, 1:100] 0 0 0 0 0 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:100] "Otu000001" "Otu000004" "Otu000002" "Otu000012" ...
##   ..$ : chr [1:100] "C460" "C171" "C2" "C267" ...
```

```r
sum(spear.ConRes4 < -0.5)
```

```
## [1] 476
```

```r
sum(spear.ConRes4 > 0.5)
```

```
## [1] 271
```

```r
pos.cor.ConRes <- which(spear.ConRes4 > 0.5)
neg.cor.ConRes <- which(spear.ConRes4 < -0.5)

# Significance Test
sig.test.neg <- list()
sig.test.pos <- list()
for (i in 1:1000) {
  # Two IndepSwap Runs
  testA <- randomizeMatrix(as.matrix(OTUsREL.dom2012), null.model = "independentswap")
  testB <- randomizeMatrix(as.matrix(resREL.dom), null.model = "independentswap")
  # Mege OTUs and Resources
  ConRes1.test <- cbind(as.matrix(testA), as.matrix(testB))
  ConRes2.test <- cbind(as.matrix(testB), as.matrix(testA))
  ConRes.test <- rbind(ConRes1.test, ConRes2.test)
  # Co-Occurence Analysis
  spear.ConRes.test <- spearman.dist(t(ConRes.test), abs = FALSE)
  spear.ConRes2.test <- spear.ConRes.test - 1
  spear.ConRes3.test <- spear.ConRes2.test
  spear.ConRes3.test[which(spear.ConRes2.test < 0.5 & spear.ConRes2.test > -0.5)] <- 0
  spear.ConRes4.test <- as.matrix(spear.ConRes3.test)[1:100, 101:200]
  which(spear.ConRes4.test < -0.5)
  which(spear.ConRes4.test > 0.5)
  sig.test.neg[[i]] <- which(spear.ConRes4.test < -0.5)
  sig.test.pos[[i]] <- which(spear.ConRes4.test > 0.5)
}
false.neg <- which((table(unlist(sig.test.neg)) / 1000) > 0.1)
false.pos <- which((table(unlist(sig.test.pos)) / 1000) > 0.1)
weak.neg <- which((table(unlist(sig.test.neg)) / 1000) > 0.05 &
                    (table(unlist(sig.test.neg)) / 1000) < 0.1)
weak.pos <- which((table(unlist(sig.test.pos)) / 1000) > 0.05 &
                    (table(unlist(sig.test.pos)) / 1000) < 0.1)

length(setdiff(neg.cor.ConRes, c(false.neg, weak.neg)))
```

```
## [1] 440
```

```r
length(setdiff(pos.cor.ConRes, c(false.pos, weak.neg)))
```

```
## [1] 245
```

```r
sum(length(setdiff(neg.cor.ConRes, c(false.neg, weak.neg))),
    length(setdiff(pos.cor.ConRes, c(false.pos, weak.neg)))) / 10000
```

```
## [1] 0.0685
```

```r
# Remove Non Significant Interactions and Define Weak Interactions
spear.ConRes5 <- spear.ConRes4
bad.ints.neg <- sort(false.neg)
bad.ints.pos <- sort(false.pos)
for (i in 1:length(bad.ints.neg)){
  if (spear.ConRes5[bad.ints.neg[i]] < 0){
    spear.ConRes5[bad.ints.neg[i]] <- 0
  } else {}
}
for (i in 1:length(bad.ints.pos)){
  if (spear.ConRes5[bad.ints.pos[i]] > 0){
    spear.ConRes5[bad.ints.pos[i]] <- 0
  } else {}
}
weak.ints.neg <- sort(weak.neg)
weak.ints.pos <- sort(weak.pos)
for (i in 1:length(weak.ints.neg)){
  if (spear.ConRes5[weak.ints.neg[i]] < 0){
    spear.ConRes5[weak.ints.neg[i]] <- -0.4
  } else {}
}
for (i in 1:length(weak.ints.pos)){
  if (spear.ConRes5[weak.ints.pos[i]] > 0){
    spear.ConRes5[weak.ints.pos[i]] <- 0.4
  } else {}
}

# Custome Color Palette
jet.colors <- colorRampPalette(c("#00007F", "blue", "#007FFF", "cyan",
                                 "#7FFF7F", "yellow", "#FF7F00", "red",
                                 "#7F0000"))
jet.colors.W <- colorRampPalette(c("#00007F", "blue", "#007FFF", "cyan",
                                   "white", "white", "white", "white",
                                   "yellow", "#FF7F00", "red", "#7F0000"))
```

Interaction Plot

```
png(filename="../figures/Figure3.png",
    width = 1800, height = 1800, res = 96*2, bg = "white")
par(opar)
ConRes.matrix <- as.matrix(spear.ConRes5)
rownames(ConRes.matrix) <- gsub("Otu000", "OTU",rownames(ConRes.matrix))
res.order <- order(colSums(ConRes.matrix <=  -0.5), decreasing = T)
cons.order <- order(rowSums(ConRes.matrix <= -0.5), decreasing = T)
order(rowSums(ConRes.matrix >= 0.5), decreasing = T)
heatmap.2(ConRes.matrix[cons.order, res.order], col = jet.colors.W, Rowv = F, Colv = F,
          dendrogram = "none", na.rm = F, na.color = "white", trace = "none",
          density.info = "none", key.xlab = "Interaction", key.title = "",
          cexCol = 0.4, cexRow = 0.4, main = "", lhei = c(2, 8), margins = c(5, 5))
mtext("Resource Availability", side = 3, line = 4, cex = 2, adj = 0.75)
mtext("Available", side = 3, line = 2, cex = 1, adj = 0.2,  padj = 1)
mtext("Restrictive", side = 3, line = 2, cex = 1, adj = 1, padj = 1)
mtext("Consumer Strategy", side = 2, line = 1, cex = 2, adj = 0)
mtext("Generalist", side = 2, line = -5, cex = 1, adj = 1, las = 1, at = 1)
mtext("Specialist", side = 2, line = -5, cex = 1, adj = 1, las = 1, at = -0.25, xpd = T)
arrows(x0 = 0.32, y0 = 1.075, x1 = 0.84, y1 = 1.075, length = 0.1, angle = 45, lwd = 3, xpd = T)
arrows(x0 = 0.08, y0 = 0.95, x1 = 0.08, y1 = -0.2, length = 0.1, angle = 45, lwd = 3, xpd = T)
dev.off() # this writes plot to folder
graphics.off() # shuts down open devices
par(opar)
```
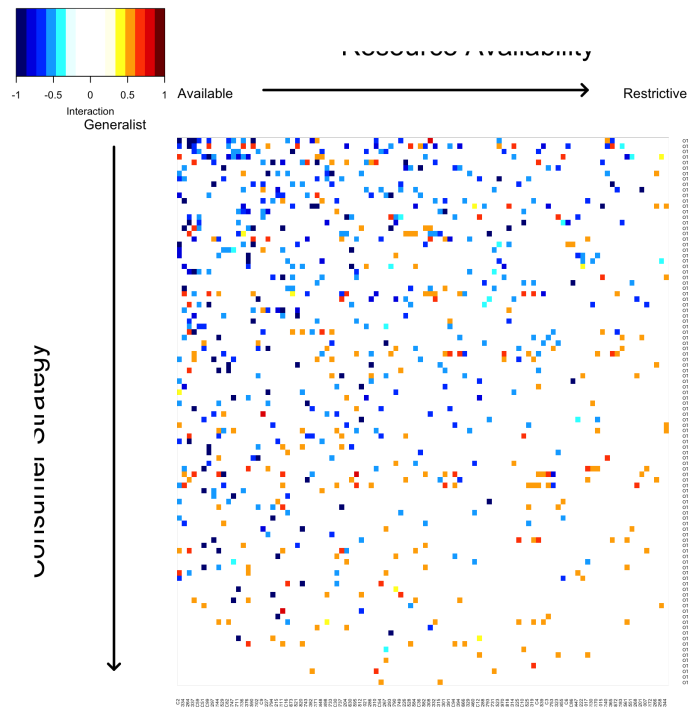

Figure 3: Interaction Heatmap

Generalist:Specialist Ratio
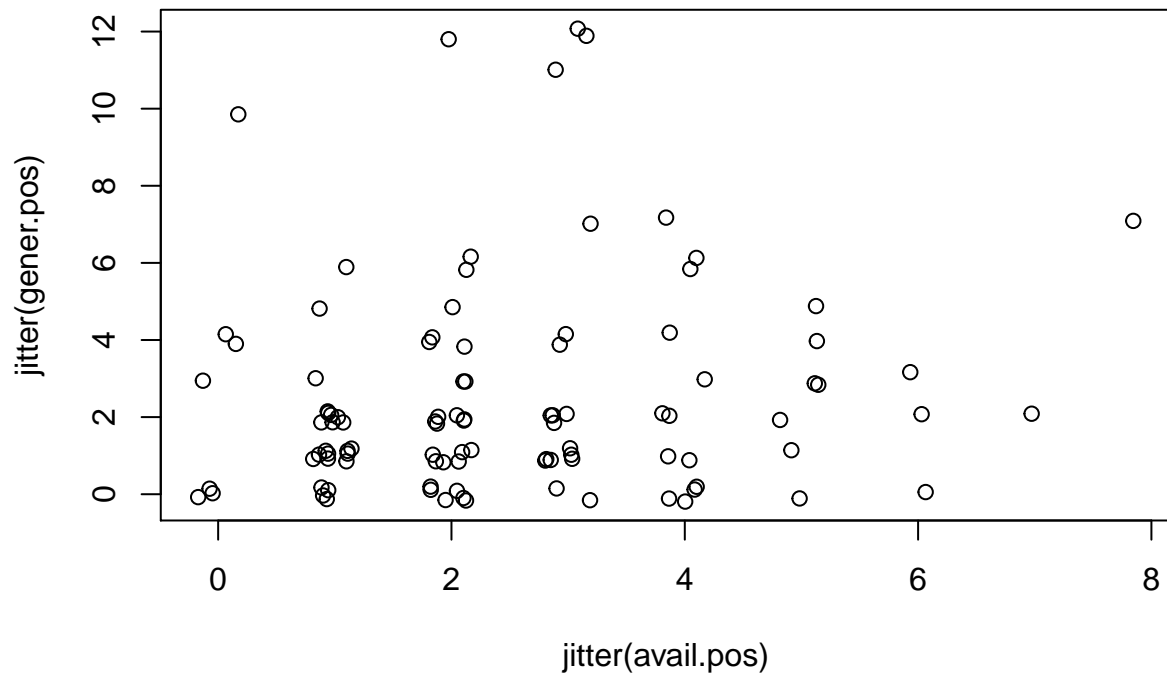
```
str(ConRes.matrix)
```

```
##  num [1:100, 1:100] 0 0 0 0 0 0.4 0 0 0 0 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:100] "OTU001" "OTU004" "OTU002" "OTU012" ...
##   ..$ : chr [1:100] "C460" "C171" "C2" "C267" ...
```

```
avail.neg <- colSums(ConRes.matrix < -0.4)
gener.neg <- rowSums(ConRes.matrix < -0.4)
avail.pos <- colSums(ConRes.matrix > 0.4)
gener.pos <- rowSums(ConRes.matrix > 0.4)

plot(jitter(avail.neg), jitter(gener.neg))
```
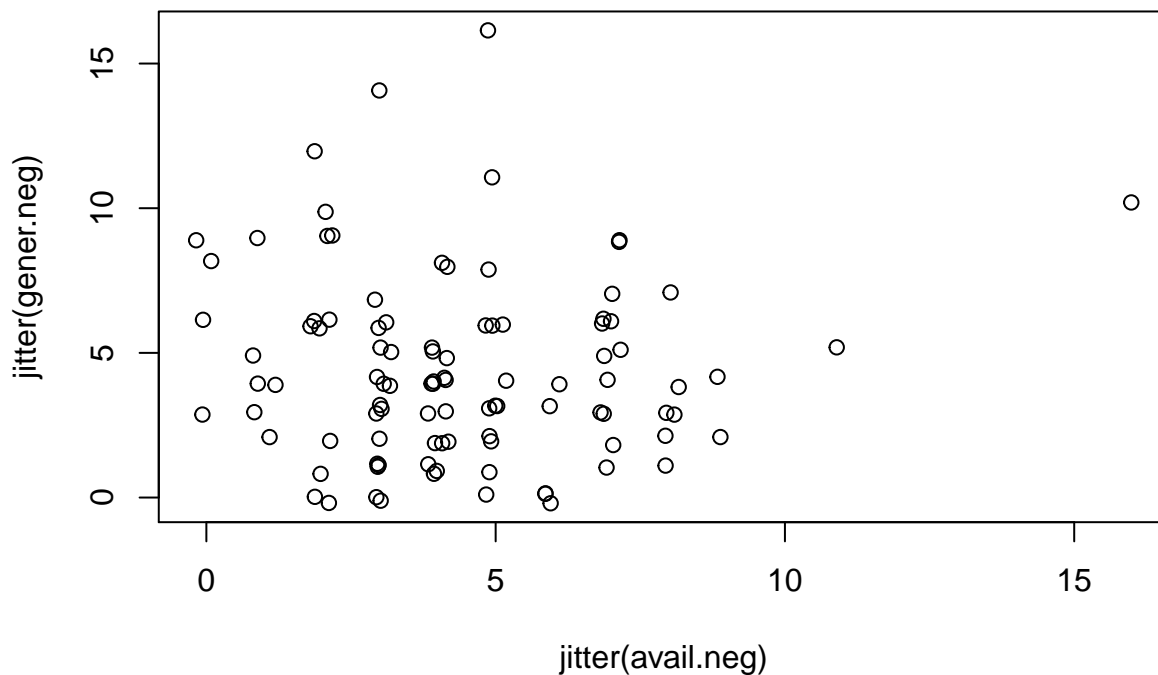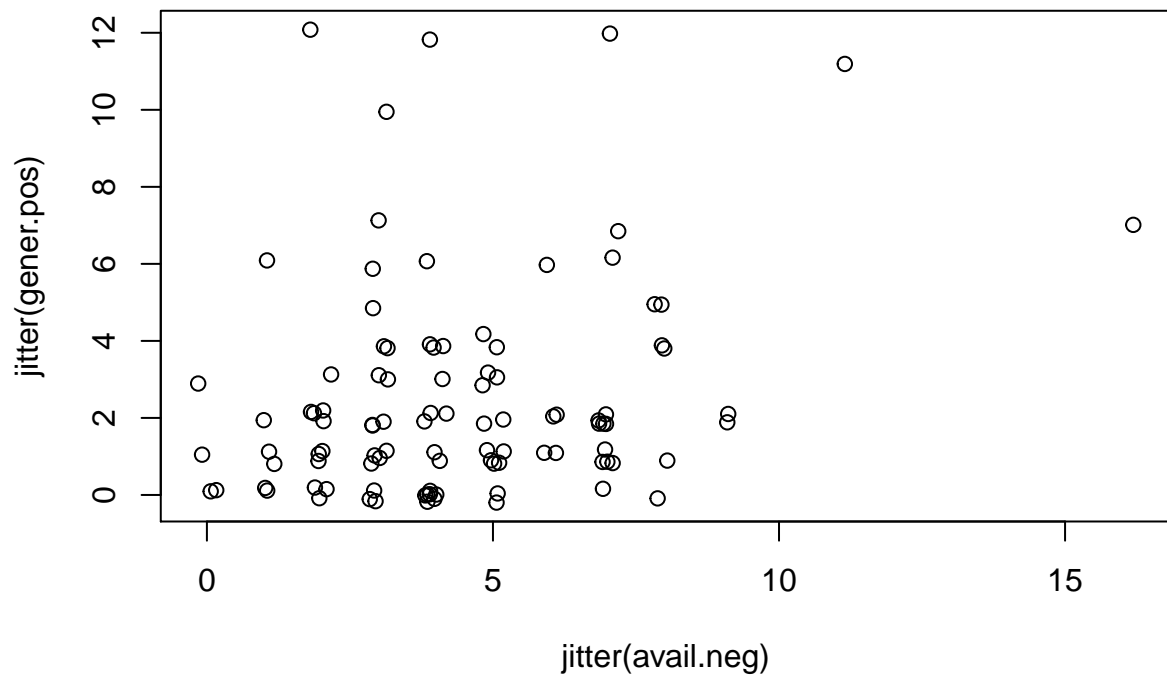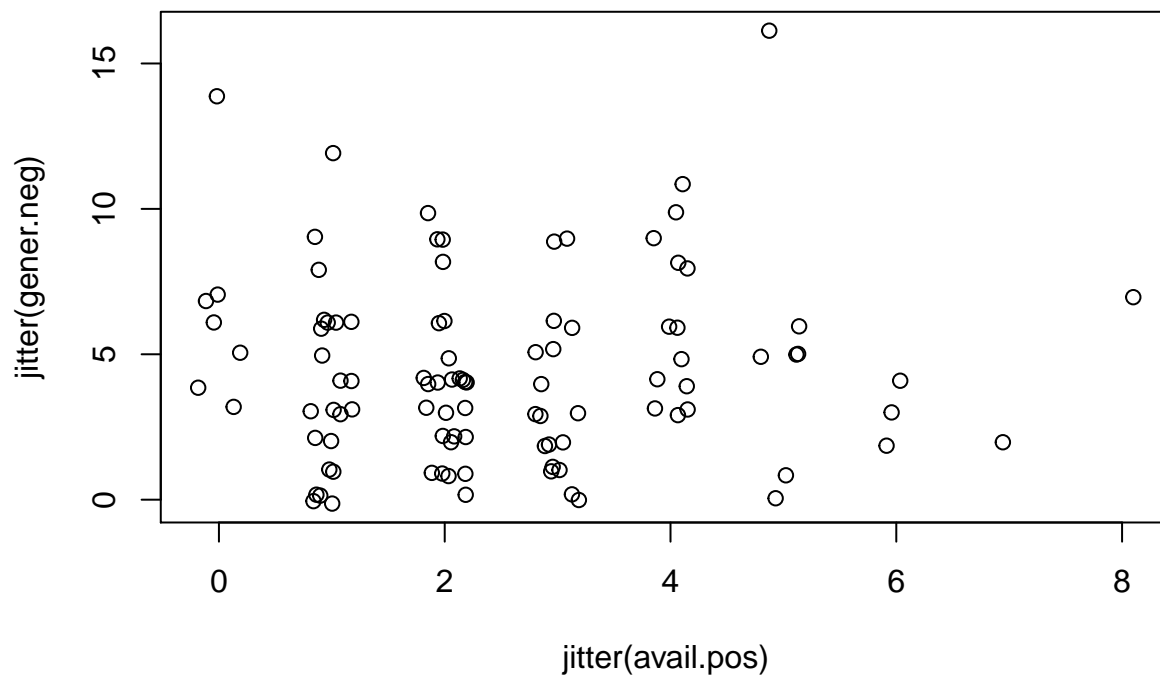


```
plot(jitter(avail.pos), jitter(gener.pos))
```
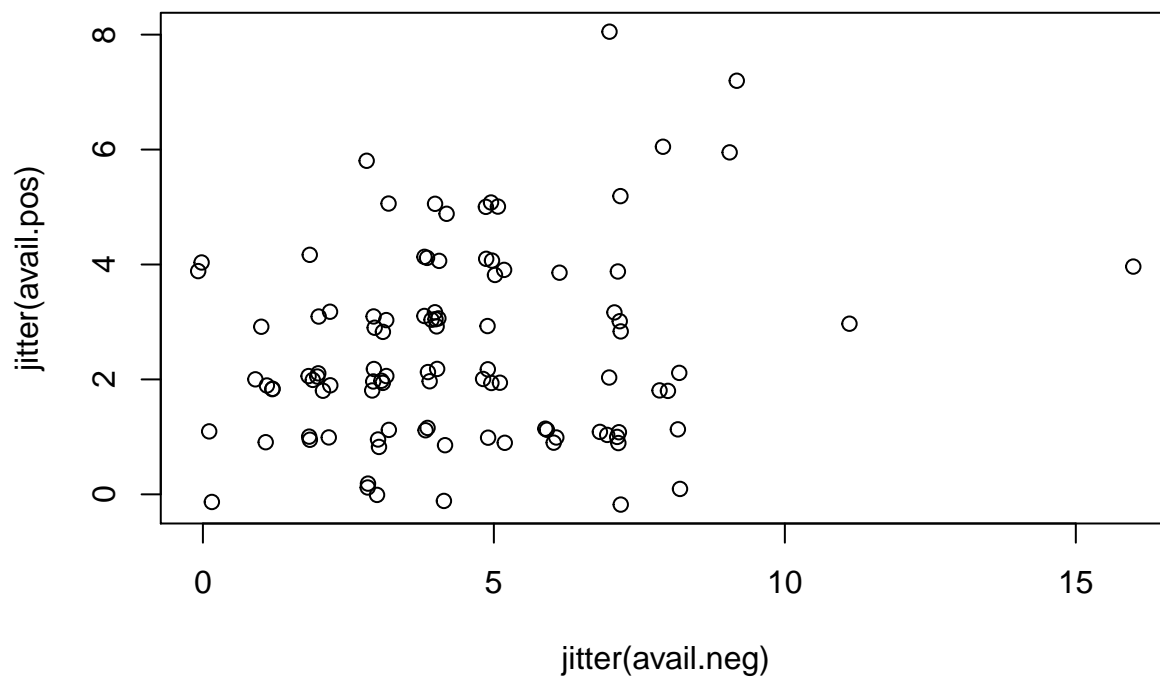
```
plot(jitter(avail.neg), jitter(gener.pos))
```
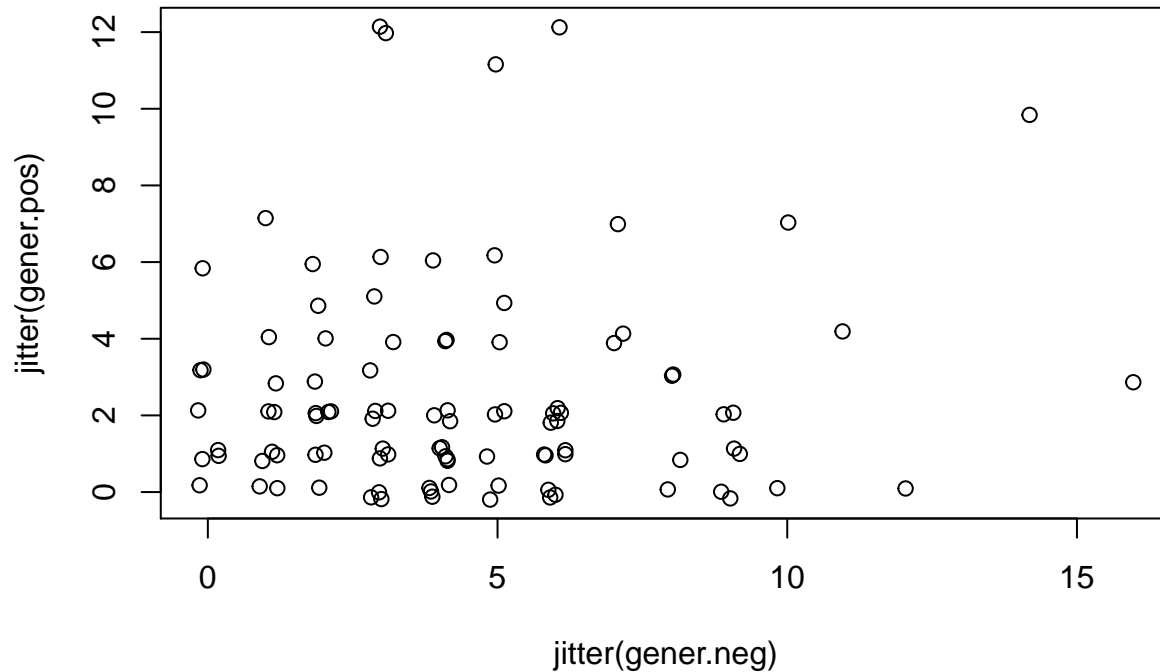


```
plot(jitter(avail.pos), jitter(gener.neg))
```

```
plot(jitter(avail.neg), jitter(avail.pos))
```



```
plot(jitter(gener.neg), jitter(gener.pos))
```

```r
specialists <- rownames(ConRes.matrix)[rowSums(ConRes.matrix < -0.4) <= 2]
generalists <- rownames(ConRes.matrix)[rowSums(ConRes.matrix < -0.4) > 2]

abundancesD <- OTUsREL.dom[design$Molecule == "DNA" & design$Year == "2012", ]
colnames(abundancesD) <- gsub("Otu000", "OTU",colnames(abundancesD))
rownames(abundancesD) <- gsub("2012_DNA", "", rownames(abundancesD))

abundancesR <- OTUsREL.dom[design$Molecule == "RNA" & design$Year == "2012", ]
colnames(abundancesR) <- gsub("Otu000", "OTU",colnames(abundancesR))
rownames(abundancesR) <- gsub("2012_RNA", "", rownames(abundancesR))

spec.matrix <- matrix(NA, nrow = 10, ncol = 4)
colnames(spec.matrix) <- c("Total.Rel.Gen", "Active.Rel.Gen",
                           "Total.Rel.Spe", "Active.Rel.Spe")
rownames(spec.matrix) <- rownames(abundancesD)

for (i in 1:(dim(spec.matrix)[1])){
  temp.gensT <- abundancesD[i, colnames(abundancesD) %in% generalists]
  temp.gensA <- abundancesR[i, colnames(abundancesR) %in% generalists]
  temp.specT <- abundancesD[i, colnames(abundancesD) %in% specialists]
  temp.specA <- abundancesR[i, colnames(abundancesR) %in% specialists]
  spec.matrix[i,1] <- round(sum(temp.gensT), 3)
  spec.matrix[i,2] <- round(sum(temp.gensA), 3)
  spec.matrix[i,3] <- round(sum(temp.specT), 3)
  spec.matrix[i,4] <- round(sum(temp.specA), 3)
}
```

Generalist:Specialists Ratio Plot

```
png(filename="../figures/Figure4.png",
    width = 1600, height =1000, res = 96*2)
par(opar)
layout(matrix(c(2,1), 2, 1), heights = c(1,5))
par(mar = c(4, 5.5, 0, 1) + 0.5)

labs <- c("Ann", "Canyon", "Howe", "Ives", "Lily", "Mountain", "Pony", "Rush",
          "Second\nPine","Upper\nPine")

# Initiate Plot
plot(spec.matrix[,1], type = "n",
     xlim = c(0.75, 10.25), ylim = c(0, 0.9),
     xaxt="n", yaxt="n", xlab = "", ylab = "")
points(seq(0.9, 9.9, 1), spec.matrix[,1], pch = 22, cex = 2,
       bg = "cornflowerblue", lwd = 2)
points(seq(1.1, 10.1, 1), spec.matrix[,2], pch = 24, cex = 2,
       bg = "cornflowerblue", lwd = 2)
points(seq(0.9, 9.9, 1), spec.matrix[,3], pch = 22, cex = 2,
       bg = "wheat3", lwd = 2)
points(seq(1.1, 10.1, 1), spec.matrix[,4], pch = 24, cex = 2,
       bg = "wheat3", lwd = 2)

abline(h = 0.5, lwd = 2, lty = 3)
axis(side=1, lwd.ticks = 2, tck=-0.02, labels = F, cex.axis = 1, at = 1:10)
axis(side=1, lwd.ticks = 2, tck=0.01, labels = F, cex.axis = 1, at = 1:10)
axis(side=2, lwd.ticks = 2, labels = T, cex.axis = 1.2, las = 1)
axis(side=2, lwd.ticks = 2, tck=0.01, labels = F, cex.axis = 1)
axis(side=3, lwd.ticks = 2, tck=-0.02, labels = F, cex.axis = 1, at = 1:10)
axis(side=3, lwd.ticks = 2, tck=0.01, labels = F, cex.axis = 1, at = 1:10)
axis(side=4, lwd.ticks = 2, tck=-0.02, labels = F, cex.axis = 1)
axis(side=4, lwd.ticks = 2, tck=0.01, labels = F, cex.axis = 1)

mtext(side = 1, text = labs, line = 1, at = seq(1:10), padj = 0.5, cex = 0.8)
mtext(side = 2, "Proportion of Community", line = 3.5, cex = 1.5)

box(lwd = 2)

# Add Legend
par(mar = c(0, 0, 0, 0) )
plot.new()

legend(0.25, 1, legend = c("Generalists (Total)", "Generalists (Active)",
                           "Specialists (Total)", "Specialists (Active)"),
       pch = c(22, 24, 22, 24), pt.lwd = 2,
       pt.bg = c("cornflowerblue", "cornflowerblue", "wheat3", "wheat3"),
       bty = "n", pt.cex = 1, cex = 1, ncol = 2, y.intersp = 1.5)


dev.off() # this writes plot to folder
```
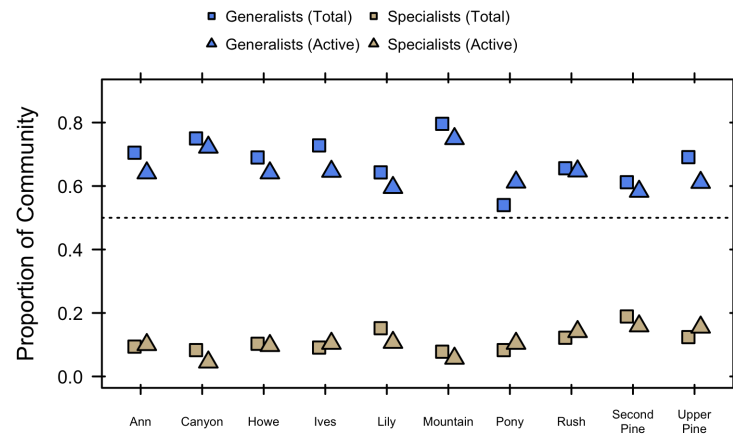
```
## pdf
```

```r
graphics.off() # shuts down open devices
par(opar)
```


Figure 4: Interaction Heatmap