Reservoir Gradient

Jay T. Lennon, Megan L. Larsen, & Mario E. Muscarella

29 October, 2015

Project looking at microbial composition and processes along a reservoir gradient.

Initial Setup

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

# Import Required Packages
require("png")
require("ggplot2")
require("maps")
require("rgdal")
require("raster")
require("OpenMx")
require("reshape")
require("ggmap")
require("grid")
require("akima")
require("plyr")
require("raster")
require("gridExtra")
```

FIGURE 1: NUTRIENT PATTERNS ACROSS DAM

```
# Load environmental data
env.dat <- read.csv("../data/ResGrad_EnvDat.csv", header = TRUE)
env.dat <- env.dat[-16,]
```

```
# ggplot theme
theme_maps <- function(base_size = 12, base_family = "Arial"){
  theme_bw(base_size = base_size, base_family = base_family) %+replace%
    theme(panel.background = element_rect(fill = "white", color = "black", size = 1.5),
          #panel.border = element_rect(color = "black"),
        #panel.margin = unit(1,1,1,1),
        panel.grid.major = element_line(colour = "white"),
        panel.grid.minor = element_line(colour = "white"),
        axis.ticks = element_line(color = "black", size = 1),
        axis.text = element_text(size = 15),
        axis.title = element_text(size = 20, face = "bold"),
        legend.position = c(0.9,0.25)
        #,axis.title.x = element_blank(),axis.title.y = element_blank()
```

```r
    )
}

# get shape files
## for map inset
usa <- map_data("usa")
IN <- map_data("state", region = "Indiana")

## for main plot
ul <- readOGR("../maps","UniversityLakePoly")
#summary(ul) # Check projection and and datum
#ul <- spTransform(ul, CRS("+proj=longlat +datum=WGS84")) # transform if necessary
ul <- fortify(ul) # raster image for plotting with ggplot2
```

```
## Regions defined for each Polygons
```

```r
# If using google map as baselayer
#ul.coords <- c(lon = -86.503087, lat = 39.188686)
#ul.map <- get_map(location = ul.coords,
#                  zoom = 17, #maptype = "terrain",
#                  source = "google", color = "bw")
#base.map <- ggmap(ul.map, extent = "device", legend = "topleft")



# Main Map
main.map <- ggplot(aes(long,lat), data = env.dat) +
  geom_polygon(fill = "grey", size = 1, color = 'black', data = ul, alpha = 1) +
  geom_point(size = 6, shape = 20) +
  theme_maps()  +
  labs(x = "\nLatitude", y = "Longitude\n")

print(main.map)

# Inset Map
inset <- ggplot() +
  theme_maps() +
  theme(axis.text = element_blank(),
        axis.ticks = element_blank(),
        panel.border = element_rect(color = NULL)) +
  labs(x = NULL, y = NULL)

inset <- inset + geom_map(data = usa, map = usa,
                    aes(x = long, y = lat, map_id = region),
                    color = "black", fill = "#e7e7e7", size = 0.15)

inset <- inset + geom_map(data = IN, map = IN,
                    aes(x = long, y=lat, map_id = region),
                    color = "black", fill = "black", size = 0.15)

inset <- inset + geom_point(aes(x = -86.503087, y = 39.188686), color = "red", size = 2)

print(inset)
```

```
# Start Plotting File
png(filename="../figures/Figure1.png",
    width = 1400, height = 1200, res = 96*2)

grid.newpage()
v1 <- viewport(width = 1, height = 1, x = 0.5, y = 0.5) #plot area for the main map
v2 <- viewport(width = 0.4, height = 0.3, x = 0.765, y = 0.31) #plot area for the inset map
print(main.map,vp = v1)
print(inset,vp = v2)

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

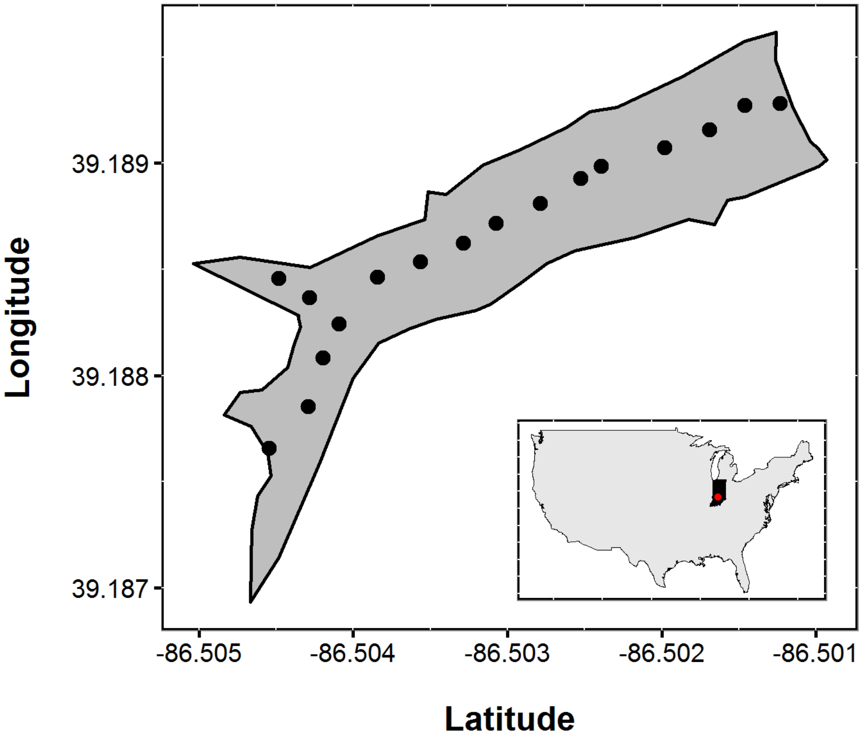

Figure 1: University Lake Map

FIGURE S1: CHEMICAL PROPERTIES

```
# Start Plotting File
png(filename="../figures/FigureS1.png",
    width = 1200, height = 1200, res = 96*2)

par(mfrow = c(2,2), oma = c(0,2,0,2)+0.5)
    #, oma = c(5, 4, 0, 0) + 0.5)
#par(mfrow = c(1,1), mar = c(1, 1, 1, 7), oma = c(5, 4, 0, 0) + 0.5)
```

```
# Total Phosphorus
TP <- plot(env.dat$dist.dam, env.dat$TP,
     ylab = "", xlab = "", cex.lab = 2, las = 1,
     ylim = c(0,140), xlim = c(-15, 400),
     pch = 22, cex = 2, bg = "white", col = "black", lwd = 2,
     yaxt = "n", xaxt = "n")
box(lwd = 2)

axis(side = 2, lwd.ticks = 2, cex.axis = 1.5, las = 1,
    labels = c("0", "40", "80", "120"), at = c(0, 40, 80, 120))

axis(side = 1, lwd.ticks = 2,  labels = F, cex.axis = 2, las = 1, mgp = c(3, 1.5, 0),
   #labels = c("0", "100", "200", "300", "400"),
   at = c(0, 100, 200, 300, 400))


mtext(expression(paste('Total Phosphorus (mg P L'^-1*')')), side = 2, line = 4, cex = 1)

# Chlorophyll
chla <- plot(env.dat$dist.dam, env.dat$chla,
     ylab = "", xlab = "", cex.lab = 2, las = 1,
     ylim = c(0,30), xlim = c(-15, 400),
     pch = 22, cex = 2, bg = "white", col = "black", lwd = 2,
     yaxt = "n", xaxt = "n")
box(lwd = 2)

axis(side = 2, lwd.ticks = 2, cex.axis = 1.5, las = 1,
    labels = c("0", "10", "20", "30"), at = c(0, 10, 20, 30))

axis(side = 1, lwd.ticks = 2, labels = F, cex.axis = 2, las = 1, mgp = c(3, 1.5, 0),
   #labels = c("0", "100", "200", "300", "400"),
   at = c(0, 100, 200, 300, 400))

mtext(expression(paste('Chlorophyll A (mg L'^-1*')')), side = 2, line = 4, cex = 1)


#Dissolved Oxygen
plot(env.dat$dist.dam, env.dat$chla,
     ylab = "", xlab = "", cex.lab = 2, las = 1,
     ylim = c(0,30), xlim = c(-15, 400),
     pch = 22, cex = 2, bg = "white", col = "black", lwd = 2,
     yaxt = "n", xaxt = "n")
box(lwd = 2)

axis(side = 2, lwd.ticks = 2, cex.axis = 1.5, las = 1,
    labels = c("0", "10", "20", "30"), at = c(0, 10, 20, 30))

axis(side = 1, lwd.ticks = 2, cex.axis = 1.5, las = 1, mgp = c(3, 1.5, 0),
   labels = c("0", "100", "200", "300", "400"),
   at = c(0, 100, 200, 300, 400))

mtext(expression(paste('Dissolved Oxygen (mg L'^-1*')')), side = 2, line = 4, cex = 1)
```

```r
#pH
plot(env.dat$dist.dam, env.dat$chla,
     ylab = "", xlab = "", cex.lab = 2, las = 1,
     ylim = c(0,30), xlim = c(-15, 400),
     pch = 22, cex = 2, bg = "white", col = "black", lwd = 2,
     yaxt = "n", xaxt = "n")
box(lwd = 2)

axis(side = 2, lwd.ticks = 2, cex.axis = 1.5, las = 1,
    labels = c("0", "10", "20", "30"), at = c(0, 10, 20, 30))

axis(side = 1, lwd.ticks = 2, cex.axis = 1.5, las = 1, mgp = c(3, 1.5, 0),
   labels = c("0", "100", "200", "300", "400"),
   at = c(0, 100, 200, 300, 400))

mtext("pH", side = 2, line = 4, cex = 1)

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

```
## pdf
##   2
```

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

FIGURE 2: METABOLISM ALONG GRADIENT

MICROBIAL METABOLISM: BP, BR, BGE

Microbial Functional Groups: Phototroph:Heterotroph

Read in data

```r
metab <- read.table("../data/res.grad.metab.txt", sep="\t", header=TRUE)
colnames(metab)[1] <- "dist"
colnames(metab)[2] <- "BP"
colnames(metab)[3] <- "BR"
BGE <- round((metab$BP/(metab$BP + metab$BR)),3)
metab <- cbind(metab, BGE)
```

```r
png(filename="../figures/Figure2.png",
    width = 1200, height = 1200, res = 96*2)

par(mfrow = c(1,1), mar = c(1, 7, 1, 7), oma = c(5, 4, 0, 0) + 0.5)
bar.layout <- layout(rbind(1, 2, 3), height = c(4, 4, 4))
#layout.show(bar.layout)

# Baterial Producivity (BP)

plot(metab$dist, metab$BP, ylab = "", xlab = "", pch = 22, ylim = c(0, 2), xlim = c(-15, 400),
     cex = 2, bg = "white", col = "black", cex.lab = 2, las = 1, lwd = 2,
     yaxt = "n", xaxt = "n")
```

```r
box(lwd = 2)

axis(side = 2, lwd.ticks = 2, cex.axis = 1.5, las = 1,
    labels = c("0.0", "1.0", "2.0"), at = c(0, 1.0, 2.0))

axis(side = 4, lwd.ticks = 2, labels = F,  cex.axis = 2, las = 1,
   at = c(0, 1, 2))

# axis(side = 1, lwd.ticks = 2, cex.axis = 1.5, las = 1,
#     labels = c("0", "100", "200", "300", "400"), at = c(0, 100, 200, 300, 400))

axis(side = 1, lwd.ticks = 2, labels = F,  cex.axis = 2, las = 1,
   at = c(0, 100, 200, 300, 400))

axis(side = 3, lwd.ticks = 2, labels = F,  cex.axis = 2, las = 1,
   at = c(0, 100, 200, 300, 400))

mtext(expression(paste('BP (', mu ,'M C h'^-1* ')')), side = 2, line = 4, cex = 1.5)

# Quadratic regression for BP
dist <- metab$dist
dist2 <- metab$dist^2
BP.fit <- lm(metab$BP ~ dist + dist2)
dist.vals <- seq(0, 375, 25)
BP.pred <- predict(BP.fit,list(dist = dist.vals, dist2 = dist.vals^2))
lines(dist.vals, BP.pred, col = "black", lwd = 2.5, lty = 6)
text(40, 1.8, labels = expression(r^2 == "0.40"), cex = 1.5)

# Bacterial Respiration (BR)

plot(metab$dist, metab$BR, ylab = "", xlab = "", pch = 22, ylim = c(0.75, 3.75), xlim = c(-15, 400),
     cex = 2, bg = "white", col = "black", cex.lab = 2, las = 1, lwd = 2,
     yaxt = "n", xaxt = "n")
box(lwd = 2)

axis(side = 2, lwd.ticks = 2, cex.axis = 1.5, las = 1,
    labels = c("1.0", "2.0", "3.0"), at = c(1, 2, 3))

axis(side = 4, lwd.ticks = 2, labels = F,  cex.axis = 2, las = 1,
   at = c(1, 2, 3))

axis(side = 1, lwd.ticks = 2, labels = F,  cex.axis = 2, las = 1,
   at = c(0, 100, 200, 300, 400))

axis(side = 3, lwd.ticks = 2, labels = F,  cex.axis = 2, las = 1,
   at = c(0, 100, 200, 300, 400))

mtext(expression(paste('BR (', mu ,'M C h'^-1* ')')), side = 2, line = 4, cex = 1.5)

# Simple linear regression for BR
BR.fit <- lm(metab$BR ~ metab$dist)
BR.int <- BR.fit$coefficients[1]
BR.slp <- BR.fit$coefficients[2]
```

```r
clip(0, 375, 0, 3.75)
abline(a = BR.int, b = BR.slp, col = "black", lwd = 2.5, lty = 6)
text(40, 3.5, labels = expression(r^2 == 0.75), cex = 1.5)

# Bacterial Growth Efficiency

plot(metab$dist, metab$BGE, ylab = "", xlab = "", pch = 22, ylim = c(0, 0.6), xlim = c(-15, 400),
     cex = 2, bg = "white", col = "black", cex.lab = 2, las = 1, lwd = 2,
     yaxt = "n", xaxt = "n")
box(lwd = 2)

axis(side = 2, lwd.ticks = 2, cex.axis = 1.5, las = 1,
    labels = c("0.0", "0.3", "0.6"), at = c(0, 0.3, 0.6))

axis(side = 4, lwd.ticks = 2, labels = F,  cex.axis = 2, las = 1,
   at = c(0, 0.3, 0.6))

axis(side = 1, lwd.ticks = 2,  cex.axis = 2, las = 1, mgp = c(3, 1.5, 0),
   labels = c("0", "100", "200", "300", "400"), at = c(0, 100, 200, 300, 400))

axis(side = 3, lwd.ticks = 2, labels = F, cex.axis = 2, las = 1,
   at = c(0, 100, 200, 300, 400))

mtext("BGE", side = 2, line = 4, cex = 1.5)
mtext("Distance (m)", side = 1, line = 4, cex = 1.5)

# Simple linear regression for BGE
BGE.fit <- lm(metab$BGE ~ metab$dist)
BGE.int <- BGE.fit$coefficients[1]
BGE.slp <- BGE.fit$coefficients[2]
clip(0, 375, 0, 0.58)
abline(a = BGE.int, b = BGE.slp, col = "black", lwd = 2.5, lty = 6)
text(40, 0.535, labels = expression(r^2 == 0.23), cex = 1.5)

# Phototroph to Heterotroph Ratio



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

FIGURE 3: Shifts in Microbial Metabolism

Crump Model: Mass Effects vs. Species Sorting

Load required R packages and tools

```r
source("../bin/MothurTools.R")
require("vegan")
```

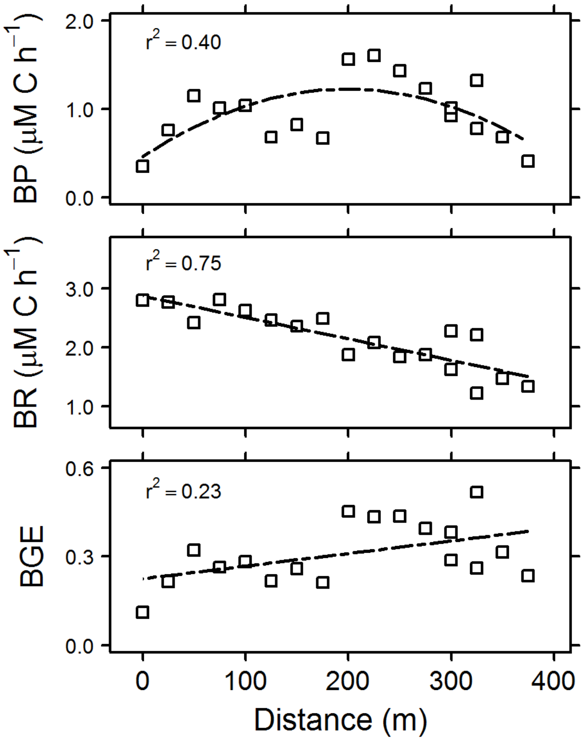

Figure 2: Microbial Processes

```
## Loading required package: vegan
## Loading required package: permute
## Loading required package: lattice
## This is vegan 2.2-1
```

```
se <- function(x, ...){sd(x, na.rm = TRUE)/sqrt(length(na.omit(x)))}
```

Import Shared and Design Files

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

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

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

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

Data Transormations

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

# Sequencing an Good's Coverage
# Sequencing Coverage
coverage <- rowSums(OTUs)

# Good's Coverage
goods <- function(x = ""){
  1 - (sum(x == 1) / rowSums(x))
}
goods.c <- goods(OTUs)

# Remove Low Coverage Samples (This code removes two sites: Site 5DNA, Site 6cDNA)
lows <- which(coverage < 10000)
lows
```

```
##  UL_05_DNA UL_06_cDNA
##         12         15
```

```r
OTUs <- OTUs[-which(coverage < 10000), ]
design <- design[-which(coverage < 10000), ]

# 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")
```

Calculate Alpha Diversity

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

# Simpson's Evenness
SimpE <- function(x = ""){
  x <- as.data.frame(x)
  D <- diversity(x, "inv")
  S <- sum((x > 0) * 1)
  E <- (D)/S
  return(E)
}
simpsE <- round(apply(OTUs, 1, SimpE), 3)

# Shannon's Diversity
H <- function(x = ""){
  x <- x[x>0]
  H = 0
  for (n_i in x){
    p = n_i / sum(x)
    H = H - p*log(p)
  }
  return(H)
}

shan <- round(apply(OTUs, 1, H), 2)
shan2 <- diversity(OTUs, index = "shannon")

alpha.div <- cbind(design, S.obs, simpsE, shan)
```

Import Phototroph

```r
# The phototrophs
cyanos.in <- "../data/UL.cyano.final.shared"
phytos.in <- "../data/UL.euks.final.shared"

cyanos <- read.otu(shared = cyanos.in, cutoff = "0.03")
phytos <- read.otu(shared = phytos.in, cutoff = "0.03")

# Remove OTUs with less than two occurences across all sites
cyanos <- cyanos[, which(colSums(cyanos) >= 2)]
phytos <- phytos[, which(colSums(phytos) >= 2)]

# Remove sites where we have low coverage
cyanos <- cyanos[-which(coverage < 10000), ]
phytos <- phytos[-which(coverage < 10000), ]

# Remove Non Intersecting Sites
ratio.sites <- intersect(intersect(rownames(cyanos), rownames(phytos)), rownames(OTUs))
cyanos <- cyanos[ratio.sites, ]
phytos <- phytos[ratio.sites, ]
heteros <- OTUs[ratio.sites, ]
design.int <- design[ratio.sites, ]

# Remove RNA Sites
DNA.samps <- which(design.int$molecule == "DNA")
cyanos <- cyanos[DNA.samps, ]
phytos <- phytos[DNA.samps, ]
heteros <- OTUs[DNA.samps, ]
design.dna <- design[DNA.samps, ]
```

```
# Observed Richness
S.cyano <- rowSums((cyanos > 0) * 1)
S.phyto <- rowSums((phytos > 0) * 1)
S.hetero <- rowSums((heteros > 0) * 1)

N.cyano <- rowSums(cyanos)
N.phyto <- rowSums(phytos)
N.hetero <- rowSums(heteros) - rowSums(cyanos)

HtoC <- N.hetero / N.cyano
HtoP <- N.hetero / N.phyto
HtoBoth <- N.hetero / (N.cyano + N.phyto)


# Ratio Across Gradient Plot
plot(HtoC ~ design.dna$distance, col= "black", pch = 22, las = 1,
     xlim = c(400, 0), ylim = c(0, 1000), cex = 1.5,
     xlab="", ylab="")
```

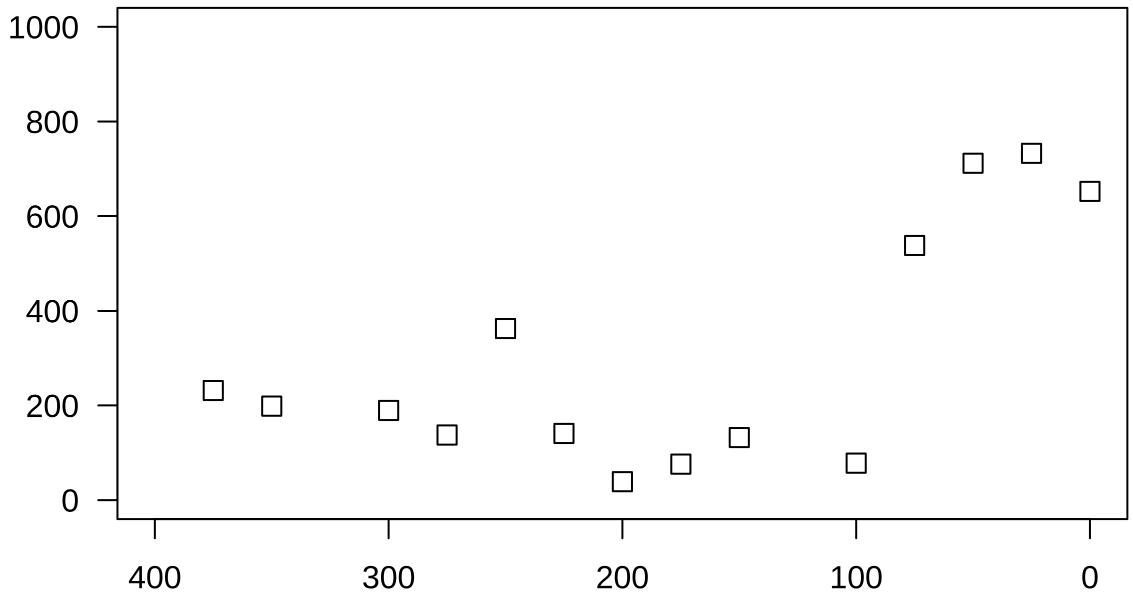

Figure 3:

Alpha Diversity Across Gradient

```
# Seperate data based on lake and soil samples
lake <- alpha.div[alpha.div$type == "water",]
soil <- alpha.div[alpha.div$type == "soil", ]

# Calculate Linear Model
model.rich <- lm(lake$S.obs ~ lake$distance * lake$molecule)
summary(model.rich)
```

```
##
## Call:
## lm(formula = lake$S.obs ~ lake$distance * lake$molecule)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -394.78 -164.65  -18.63  122.24  722.23
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    368.9165   125.7785   2.933  0.00637 **
## lake$distance                    4.4396     0.5291   8.390 2.30e-09 ***
## lake$moleculeRNA               113.2278   176.7261   0.641  0.52658
## lake$distance:lake$moleculeRNA  -4.4788     0.7445  -6.016 1.33e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 253.6 on 30 degrees of freedom
## Multiple R-squared:  0.8407, Adjusted R-squared:  0.8247
## F-statistic: 52.76 on 3 and 30 DF,  p-value: 4.472e-12
```

```
# Calculate Confidance Intervals of Model
newdata.rich <- data.frame(cbind(lake$molecule, lake$distance))
conf95.rich <- predict(model.rich, newdata.rich, interval="confidence")

# Average Richness in Terrestrial Habitat
mean(soil$S.obs)
```

```
## [1] 7158.667
```

```
# Dummy Variables Regression Model ("Species Richness"")
D1 <- (lake$molecule == "RNA")*1
fit.Fig.3a <- lm(lake$S.obs ~ lake$distance + D1 + lake$distance*D1)
summary(fit.Fig.3a)
```

```
##
## Call:
## lm(formula = lake$S.obs ~ lake$distance + D1 + lake$distance *
##     D1)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -394.78 -164.65  -18.63  122.24  722.23
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       368.9165   125.7785   2.933  0.00637 **
## lake$distance       4.4396     0.5291   8.390 2.30e-09 ***
## D1                113.2278   176.7261   0.641  0.52658
## lake$distance:D1   -4.4788     0.7445  -6.016 1.33e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 253.6 on 30 degrees of freedom
## Multiple R-squared:  0.8407, Adjusted R-squared:  0.8247
## F-statistic: 52.76 on 3 and 30 DF,  p-value: 4.472e-12
```

```
DNA.int.3a <- fit.Fig.3a$coefficients[1]
DNA.slp.3a <- fit.Fig.3a$coefficients[2]
RNA.int.3a <- DNA.int.3a + fit.Fig.3a$coefficients[3]
RNA.slp.3a <- DNA.slp.3a + fit.Fig.3a$coefficients[4]
```

Similarity To Terrestrial Habitat Across Gradient

```
# Similarity to Soil Sample
UL.bray      <- 1 - as.matrix(vegdist(OTUsREL.log, method="bray"))
UL.bray.lake <- UL.bray[-c(1:3), 1:3]
bray.mean    <- round(apply(UL.bray.lake, 1, mean), 3)
bray.se      <- round(apply(UL.bray.lake, 1, se), 3)
UL.sim       <- cbind(design[-c(1:3), ], bray.mean, bray.se)

# Calculate Linear Model
model.terr <- lm(UL.sim$bray.mean ~ UL.sim$distance * UL.sim$molecule)
summary(model.terr)
```

```
##
## Call:
## lm(formula = UL.sim$bray.mean ~ UL.sim$distance * UL.sim$molecule)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.051051 -0.012638 -0.002573  0.008963  0.091666
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         1.567e-02  1.461e-02   1.073 0.291795
## UL.sim$distance                     4.143e-04  6.144e-05   6.743 1.78e-07
## UL.sim$moleculeRNA                  1.127e-02  2.052e-02   0.549 0.586965
## UL.sim$distance:UL.sim$moleculeRNA -3.855e-04  8.646e-05  -4.459 0.000107
##
## (Intercept)
## UL.sim$distance                    ***
## UL.sim$moleculeRNA
## UL.sim$distance:UL.sim$moleculeRNA ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02945 on 30 degrees of freedom
## Multiple R-squared:  0.754,  Adjusted R-squared:  0.7294
## F-statistic: 30.65 on 3 and 30 DF,  p-value: 2.868e-09
```

```
# Calculate Confidance Intervals of Model
newdata.terr <- data.frame(cbind(UL.sim$molecule, UL.sim$distance))
conf95.terr <- predict(model.terr, newdata.terr, interval="confidence")

# Dummy Variables Regression Model ("Terrestrial Influence")
D2 <- (UL.sim$molecule == "RNA")*1
fit.Fig.3b <- lm(UL.sim$bray.mean ~ UL.sim$distance + D2 + UL.sim$distance*D2)
summary(fit.Fig.3b)
```

```
##
## Call:
## lm(formula = UL.sim$bray.mean ~ UL.sim$distance + D2 + UL.sim$distance *
##     D2)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.051051 -0.012638 -0.002573  0.008963  0.091666
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1.567e-02  1.461e-02   1.073 0.291795
## UL.sim$distance     4.143e-04  6.144e-05   6.743 1.78e-07 ***
## D2                  1.127e-02  2.052e-02   0.549 0.586965
## UL.sim$distance:D2 -3.855e-04  8.646e-05  -4.459 0.000107 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02945 on 30 degrees of freedom
## Multiple R-squared:  0.754,  Adjusted R-squared:  0.7294
## F-statistic: 30.65 on 3 and 30 DF,  p-value: 2.868e-09
```

```
DNA.int.3b <- fit.Fig.3b$coefficients[1]
DNA.slp.3b <- fit.Fig.3b$coefficients[2]
RNA.int.3b <- DNA.int.3b + fit.Fig.3b$coefficients[3]
RNA.slp.3b <- DNA.slp.3b + fit.Fig.3b$coefficients[4]
```

Similarity To Lake Habitat Across Gradient

```
# Similarity to Lake Sample 1
UL.bray2      <- 1 - as.matrix(vegdist(OTUsREL.log, method="bray"))
UL.bray.lake2 <- UL.bray[-c(1:3), 4:7]
UL.sim2       <- cbind(design[-c(1:3), ], "DNA" = apply(UL.bray.lake2[,c(1,3)], 1, mean),
                                          "RNA" = apply(UL.bray.lake2[,c(2,4)], 1, mean))

# Calculate Linear Model
model.lake1 <- lm(UL.sim2$DNA ~ UL.sim2$distance * UL.sim2$molecule)
model.lake2 <- lm(UL.sim2$RNA ~ UL.sim2$distance * UL.sim2$molecule)
summary(model.lake1)
```

```
##
## Call:
## lm(formula = UL.sim2$DNA ~ UL.sim2$distance * UL.sim2$molecule)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.212825 -0.075949 -0.006199  0.054511  0.254650
##
## Coefficients:
##                                     Estimate Std. Error t value
## (Intercept)                        0.7804831  0.0493547  15.814
## UL.sim2$distance                  -0.0015905  0.0002076  -7.660
## UL.sim2$moleculeRNA               -0.4639770  0.0693462  -6.691
## UL.sim2$distance:UL.sim2$moleculeRNA  0.0014089  0.0002921   4.823
##                                   Pr(>|t|)
## (Intercept)                       4.27e-16 ***
## UL.sim2$distance                  1.52e-08 ***
## UL.sim2$moleculeRNA               2.06e-07 ***
## UL.sim2$distance:UL.sim2$moleculeRNA 3.84e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09951 on 30 degrees of freedom
## Multiple R-squared:  0.7385, Adjusted R-squared:  0.7124
## F-statistic: 28.24 on 3 and 30 DF,  p-value: 7.107e-09
```

```
summary(model.lake2)
```

```
##
## Call:
## lm(formula = UL.sim2$RNA ~ UL.sim2$distance * UL.sim2$molecule)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.278785 -0.037188  0.002748  0.040844  0.290619
##
## Coefficients:
##                                     Estimate Std. Error t value
## (Intercept)                        4.249e-01  5.839e-02   7.276
## UL.sim2$distance                  -7.120e-04  2.456e-04  -2.898
## UL.sim2$moleculeRNA                1.850e-02  8.205e-02   0.226
## UL.sim2$distance:UL.sim2$moleculeRNA -3.571e-05  3.457e-04  -0.103
##                                   Pr(>|t|)
## (Intercept)                       4.22e-08 ***
## UL.sim2$distance                   0.00695 **
## UL.sim2$moleculeRNA                   0.82311
## UL.sim2$distance:UL.sim2$moleculeRNA  0.91840
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1177 on 30 degrees of freedom
## Multiple R-squared:  0.3743, Adjusted R-squared:  0.3117
## F-statistic: 5.982 on 3 and 30 DF,  p-value: 0.002539
```

```
# Calculate Confidance Intervals of Model
newdata.lake <- data.frame(cbind(UL.sim2$molecule, UL.sim2$distance))
conf95.lake <- predict(model.lake1, newdata.lake, interval="confidence")

# Dummy Variables Regression Model ("Lake Influence")
D3 <- (UL.sim2$molecule == "RNA")*1
fit.Fig.3c <- lm(UL.sim2$DNA ~ UL.sim2$distance + D3 + UL.sim2$distance*D3)
summary(fit.Fig.3c)
```

```
##
## Call:
## lm(formula = UL.sim2$DNA ~ UL.sim2$distance + D3 + UL.sim2$distance *
##     D3)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.212825 -0.075949 -0.006199  0.054511  0.254650
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.7804831  0.0493547  15.814 4.27e-16 ***
## UL.sim2$distance    -0.0015905  0.0002076  -7.660 1.52e-08 ***
## D3                  -0.4639770  0.0693462  -6.691 2.06e-07 ***
## UL.sim2$distance:D3  0.0014089  0.0002921   4.823 3.84e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09951 on 30 degrees of freedom
## Multiple R-squared:  0.7385, Adjusted R-squared:  0.7124
## F-statistic: 28.24 on 3 and 30 DF,  p-value: 7.107e-09
```

```
DNA.int.3c <- fit.Fig.3c$coefficients[1]
DNA.slp.3c <- fit.Fig.3c$coefficients[2]
RNA.int.3c <- DNA.int.3c + fit.Fig.3c$coefficients[3]
RNA.slp.3c <- DNA.slp.3c + fit.Fig.3c$coefficients[4]
```

Figure 3 Plot

```
# Define Plot Parameters
opar <- par()
# par(mar = c(5,6, 1, 1))
mol <- rep(NA, length(lake$molecule))
```

```
  for (i in 1:length(mol)){
    if (lake$molecule[i] == "DNA"){
      mol[i] <- 22
    } else {
      mol[i] <- 24
    }
  }
cols <- rep(NA, length(lake$molecule))
  for (i in 1:length(cols)){
    if (lake$molecule[i] == "DNA"){
      cols[i] <- "gray15"
    } else {
      cols[i] <- "gray75"
    }
  }

# Initial Plot
png(filename="../figures/Figure3.png",
    width = 1200, height =1200, res = 96*2)

par(mfrow = c(1,1), mar = c(1, 7, 1, 7), oma = c(5, 4, 0, 0) + 0.5)
bar.layout <- layout(rbind(1, 2, 3), height = c(4, 4, 4))

# Richness Across Gradient Plot
plot(lake$S.obs ~ lake$distance, col= "black", bg = cols, pch=mol, las = 1,
     xlim = c(400, 0), ylim = c(0, 2750), cex = 1.5,
     xlab="", ylab="", xaxt="n")

#   matlines(lake$distance[lake$molecule == "DNA"], conf95.rich[lake$molecule == "DNA", ],
#          lty = c(1, 0, 0), col=c("black", "gray50", "gray50"), lwd=c(2, 1, 1))
#   matlines(lake$distance[lake$molecule == "RNA"], conf95.rich[lake$molecule == "RNA", ],
#          lty = c(1, 0, 0), col=c("black", "gray50", "gray50"), lwd=c(2, 1, 1))

# Add multiple regression lines
clip(400, 0, 0, 2750)
abline(a = DNA.int.3a, b = DNA.slp.3a, col = "black", lwd = 2.5, lty = 6)

clip(400, 0, 0, 2750)
abline(a = RNA.int.3a, b = RNA.slp.3a, col = "black", lwd = 2.5, lty = 4)

  axis(side = 1, lwd.ticks = 2, labels = F, cex.axis = 1, las = 1)
  axis(side = 2, lwd.ticks = 2, cex.axis = 1, las = 1)
  axis(side = 3, lwd.ticks = 2, tck=-0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 4, lwd.ticks = 2, tck=-0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 1, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 2, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 3, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 4, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)

  # mtext("Distance (m)" , side = 1, line = 3, cex=1.5)
  mtext("Richness\n(S)" , side = 2, line = 4, cex=1.5)

  legend("topright", legend = levels(lake$molecule), pch=c(22, 24),
       pt.bg = c("gray15", "gray75"), bty='n', cex = 1.25)

  box(lwd=2)

# Terrestrial Influence Plot
#   mol <- rep(NA, length(UL.sim$molecule))
#     for (i in 1:length(mol)){
#       if (UL.sim$molecule[i] == "DNA"){
#         mol[i] <- 21
#       } else {
#         mol[i] <- 24
#       }
#     }
#   cols <- rep(NA, length(UL.sim$molecule))
#     for (i in 1:length(cols)){
#       if (UL.sim$molecule[i] == "DNA"){
#         cols[i] <- "gray15"
#       } else {
#         cols[i] <- "gray75"
#       }
#     }

plot(UL.sim$bray.mean ~ UL.sim$distance, col= "black", bg = cols, pch=mol, las = 1,
     xlim = c(400, 0), ylim = c(0, 0.25), cex = 1.5,
     xlab="", ylab="", xaxt="n")

#   matlines(lake$distance[lake$molecule == "DNA"], conf95.terr[lake$molecule == "DNA", ],
#          lty = c(1, 0, 0), col=c("black", "gray50", "gray50"), lwd=c(2, 1, 1))
#   matlines(lake$distance[lake$molecule == "RNA"], conf95.terr[lake$molecule == "RNA", ],
#          lty = c(1, 0, 0), col=c("black", "gray50", "gray50"), lwd=c(2, 1, 1))

# Add multiple regression lines
clip(400, 0, 0, 0.27)
abline(a = DNA.int.3b, b = DNA.slp.3b, col = "black", lwd = 2.5, lty = 6)

clip(400, 0, 0, 0.27)
abline(a = RNA.int.3b, b = RNA.slp.3b, col = "black", lwd = 2.5, lty = 4)

  axis(side = 1, lwd.ticks = 2, labels = F, cex.axis = 1, las = 1)
  axis(side = 2, lwd.ticks = 2, cex.axis = 1, las = 1)
  axis(side = 3, lwd.ticks = 2, tck=-0.05, labels = F,  cex.axis = 2, las = 1)
  axis(side = 4, lwd.ticks = 2, tck=-0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 1, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 2, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 3, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 4, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)

  # mtext("Distance (m)" , side = 1, line = 3, cex=1.5)
  mtext("Terrestrial\nInfluence" , side = 2, line = 4, cex=1.5)

  legend("topright", legend = levels(UL.sim$molecule), pch=c(22, 24),
       pt.bg = c("gray15", "gray75"), bty='n', cex = 1.25)
  box(lwd=2)

# Lake Influence Plot
plot(UL.sim2$DNA ~ UL.sim2$distance, col= "black", bg = cols, pch=mol, las = 1,
     xlim = c(400, 0), ylim = c(0, 1), cex = 1.5,
     xlab="", ylab="")

#   matlines(lake$distance[lake$molecule == "DNA"], conf95.lake[lake$molecule == "DNA", ],
#          lty = c(1, 0, 0), col=c("black", "gray50", "gray50"), lwd=c(2, 1, 1))
#   matlines(lake$distance[lake$molecule == "RNA"], conf95.lake[lake$molecule == "RNA", ],
#          lty = c(1, 0, 0), col=c("black", "gray50", "gray50"), lwd=c(2, 1, 1))

# Add multiple regression lines
clip(400, 0, 0, 1)
abline(a = DNA.int.3c, b = DNA.slp.3c, col = "black", lwd = 2.5, lty = 6)

clip(400, 0, 0, 1)
abline(a = RNA.int.3c, b = RNA.slp.3c, col = "black", lwd = 2.5, lty = 4)

  axis(side = 1, lwd.ticks = 2, cex.axis = 1, las = 1)
  axis(side = 2, lwd.ticks = 2, cex.axis = 1, las = 1)
  axis(side = 3, lwd.ticks = 2, tck=-0.05, labels = F,  cex.axis = 2, las = 1)
  axis(side = 4, lwd.ticks = 2, tck=-0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 1, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 2, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 3, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 4, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)

  mtext("Distance (m)" , side = 1, line = 3, cex=1.5)
  mtext("Lake\nInfluence", side = 2, line = 4, cex=1.5)

  legend("topleft", legend = levels(UL.sim$molecule), pch=c(22, 24),
       pt.bg = c("gray15", "gray75"), bty='n', cex = 1.25)

  box(lwd=2)



# Close Plot Defice
dev.off()
```

```
## pdf
##   2
```

```
graphics.off()
```

Identifying the Soil Bacteria

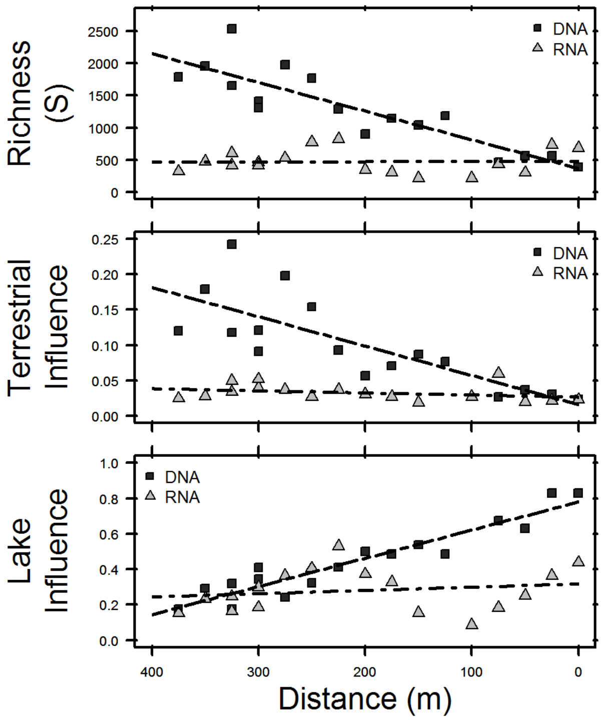

Figure 4: Microbial Community Shifts

```
soil.only <- OTUs[, which(colSums(OTUs[-c(1:3),]) == 0)]
lake.n.soil <- OTUs[, setdiff(colnames(OTUs),colnames(soil.only))]

w.dna <- OTUs[which(design$molecule == "DNA" & design$type == "water"), ]
w.rna <- OTUs[which(design$molecule == "RNA" & design$type == "water"), ]

nvr.act <- which(colSums(w.rna) == 0)

terr.lake <- w.dna[ , c(names(nvr.act))]

terr.rich <- rowSums((terr.lake > 0) * 1)

terr.REL <- rowSums(terr.lake) / rowSums(w.dna)

design.dna <- design[which(design$molecule == "DNA" & design$type == "water"), ]

terr.rich.log <- log10(terr.rich)
terr.REL.log <- log10(terr.REL)

terr.mod1 <- lm(terr.rich.log ~ design.dna$distance)
terr.mod2 <- lm(terr.REL.log ~ design.dna$distance)

summary(terr.mod1)
```

```
##
```

```
## Call:
## lm(formula = terr.rich.log ~ design.dna$distance)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.24705 -0.12361  0.02547  0.09541  0.22716
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)         2.025051   0.077485  26.135 6.35e-14 ***
## design.dna$distance 0.003018   0.000326   9.257 1.37e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1562 on 15 degrees of freedom
## Multiple R-squared:  0.851,  Adjusted R-squared:  0.8411
## F-statistic:  85.7 on 1 and 15 DF,  p-value: 1.365e-07
```

```
summary(terr.mod2)
```

```
##
## Call:
## lm(formula = terr.REL.log ~ design.dna$distance)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43842 -0.10220 -0.00186  0.10962  0.42941
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -2.147129   0.113388  -18.94 6.96e-12 ***
## design.dna$distance  0.002900   0.000477    6.08 2.10e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2286 on 15 degrees of freedom
## Multiple R-squared:  0.7114, Adjusted R-squared:  0.6921
## F-statistic: 36.97 on 1 and 15 DF,  p-value: 2.105e-05
```

```
# # Calculate Confidance Intervals of Model
# newdata.lake <- data.frame(cbind(UL.sim2$molecule, UL.sim2$distance))
# conf95.lake <- predict(terr.mod1, newdata.lake, interval="confidence")
#
# # Dummy Variables Regression Model ("Lake Influence")
# D3 <- (UL.sim2$molecule == "RNA")*1
# fit.Fig.3c <- lm(UL.sim2$DNA ~ UL.sim2$distance + D3 + UL.sim2$distance*D3)
# summary(fit.Fig.3c)
#
# DNA.int.3c <- fit.Fig.3c$coefficients[1]
# DNA.slp.3c <- fit.Fig.3c$coefficients[2]
# RNA.int.3c <- DNA.int.3c + fit.Fig.3c$coefficients[3]
# RNA.slp.3c <- DNA.slp.3c + fit.Fig.3c$coefficients[4]
```

Figure 4: Soil Organisms Plot

```
# Initial Plot
png(filename="../figures/Figure4.png",
    width = 1200, height =1200, res = 96*2)

par(mfrow = c(1,1), mar = c(1, 5, 1, 1), oma = c(4, 4, 0, 0) + 0.5)
bar.layout <- layout(rbind(1, 2), height = c(4, 4))

# Soil OTU Richness Across Gradient Plot
plot(terr.rich.log ~ design.dna$distance, col= "black", pch=22, las = 1,
     xlim = c(400, 0), ylim = c(1.5, 3.5), cex = 1.5,
     xlab="", ylab="", xaxt="n", yaxt="n")

  axis(side = 1, lwd.ticks = 2, labels = T, cex.axis = 1, las = 1)
  axis(side = 1, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 3, lwd.ticks = 2, tck=-0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 3, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 2, lwd.ticks = 2, at = c(2, 3), labels = c(10^2, 10^3), cex.axis = 1, las = 1)
  axis(side = 2, lwd.ticks = 2, tck = -0.02, at = log10(c(seq(10, 100, by = 10),
       seq(100, 1000, by = 100), seq(1000, 10000, by = 1000))), labels = F, cex.axis = 1, las = 1)
  axis(side = 2, lwd.ticks = 2, at = c(2, 3), tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 2, lwd.ticks = 2, tck = 0.005, at = log10(c(seq(10, 100, by = 10),
       seq(100, 1000, by = 100), seq(1000, 10000, by = 1000))), labels = F, cex.axis = 1, las = 1)
  axis(side = 4, lwd.ticks = 2, at = c(2, 3), tck=-0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 4, lwd.ticks = 2, at = c(2, 3), tck=0.02, labels = F,  cex.axis = 2, las = 1)
  axis(side = 4, lwd.ticks = 2, tck = 0.01, at = log10(c(seq(10, 100, by = 10),
       seq(100, 1000, by = 100), seq(1000, 10000, by = 1000))), labels = F, cex.axis = 1, las = 1)

  # mtext("Distance (m)" , side = 1, line = 3, cex=1.5)
  mtext("Transient\nRichness\n(S)" , side = 2, line = 4, cex=1.5)
  box(lwd=2)

#   # Add multiple regression lines
#   clip(400, 0, 0, 1)
#   abline(a = DNA.int.3c, b = DNA.slp.3c, col = "black", lwd = 2.5, lty = 6)
#
#   clip(400, 0, 0, 1)
#   abline(a = RNA.int.3c, b = RNA.slp.3c, col = "black", lwd = 2.5, lty = 4)


# Soil OTU Relative Abundance Across Gradient Plot
plot(terr.REL.log ~ design.dna$distance, col= "black", pch=22, las = 1,
     xlim = c(400, 0), ylim = c(-2.5, -.5), cex = 1.5,
     xlab="", ylab="", xaxt="n", yaxt="n")

  axis(side = 1, lwd.ticks = 2, labels = T, cex.axis = 1, las = 1)
  axis(side = 1, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 3, lwd.ticks = 2, tck=-0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 3, lwd.ticks = 2, tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 2, lwd.ticks = 2, at = c(-2, -1), labels = c(0.01, 0.1), cex.axis = 1, las = 1)
  axis(side = 2, lwd.ticks = 2, tck = -0.02, at = log10(c(seq(0.001, 0.01, by = 0.001),
       seq(0.01, 0.1, by = 0.01), seq(0.1, 1, by = 0.1))), labels = F, cex.axis = 1, las = 1)
  axis(side = 2, lwd.ticks = 2, at = c(-2, -1), tck=0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 2, lwd.ticks = 2, tck = 0.005, at = log10(c(seq(0.001, 0.01, by = 0.001),
       seq(0.01, 0.1, by = 0.01), seq(0.1, 1, by = 0.1))), labels = F, cex.axis = 1, las = 1)
  axis(side = 4, lwd.ticks = 2, at = c(-2, -1), tck=-0.01, labels = F,  cex.axis = 2, las = 1)
  axis(side = 4, lwd.ticks = 2, at = c(-2, -1), tck=0.02, labels = F,  cex.axis = 2, las = 1)
  axis(side = 4, lwd.ticks = 2, tck = 0.01, at = log10(c(seq(0.001, 0.01, by = 0.001),
       seq(0.01, 0.1, by = 0.01), seq(0.1, 1, by = 0.1))), labels = F, cex.axis = 1, las = 1)


  mtext("Distance (m)" , side = 1, line = 3, cex=1.5)
  mtext("Transient\nRelative\nAbundance", side = 2, line = 4, cex=1.5)

  box(lwd=2)


# Close Plot Defice
dev.off()
```

```
## pdf
##   2
```

```
graphics.off()
```

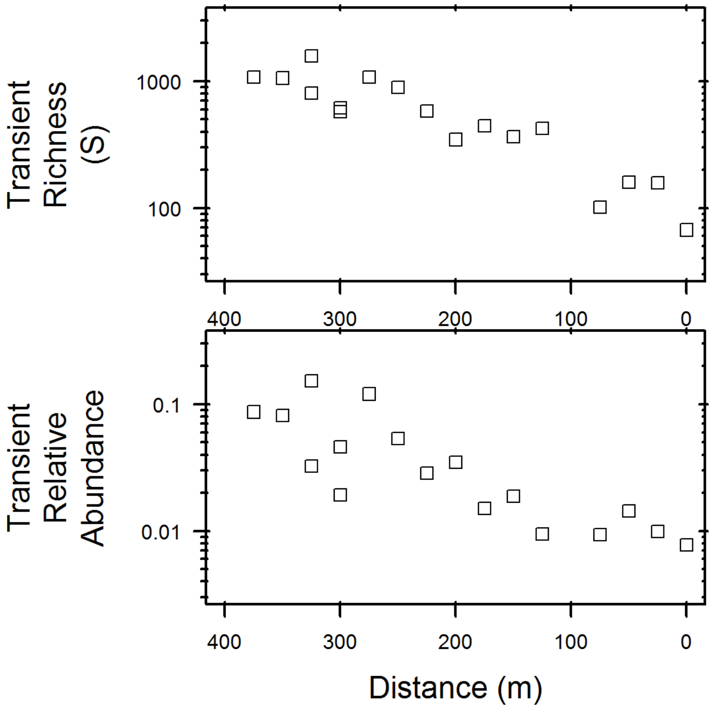

Figure 5: Transient Species Distributions