Dormancy and dispersal structure bacterial communities across ecosystem boundaries

Nathan I. Wisnoski, Mario E. Muscarella, Megan L. Larsen, and Jay T. Lennon

20 December, 2019

Initial Setup

First, we'll load the packages we'll need for the analysis, as well as some other functions.

```r
# Import Required Packages
library("png")
library("grid")
library("tidyverse")
library("vegan")
library("viridis")
library("cowplot")
library("ggrepel")
library("iNEXT")
library("broom")
library("ggpmisc")
library("pander")
library("lubridate")
library("betapart")
library("adespatial")
library("VennDiagram")

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

Next, we'll set the aesthetics of the figures we will produce.

```r
my.cols <- RColorBrewer::brewer.pal(n = 4, name = "Greys")[3:4]

# Set theme for figures in the paper
theme_set(theme_classic() +
  theme(axis.title = element_text(size = 16),
        axis.title.x = element_text(margin = margin(t = 15, b = 15)),
        axis.title.y = element_text(margin = margin(l = 15, r = 15)),
        axis.text = element_text(size = 14),
        axis.text.x = element_text(margin = margin(t = 5)),
        axis.text.y = element_text(margin = margin(r = 5)),
        #axis.line.x = element_line(size = 1),
        #axis.line.y = element_line(size = 1),
        axis.line.x = element_blank(),
        axis.line.y = element_blank(),
        axis.ticks.x = element_line(size = 1),
        axis.ticks.y = element_line(size = 1),
        axis.ticks.length = unit(.1, "in"),
        panel.border = element_rect(color = "black", fill = NA, size = 1.5),
        legend.title = element_blank(),
        legend.text = element_text(size = 14),
        strip.text = element_text(size = 14),
        strip.background = element_blank()
        ))
```

Import Data

Here, we read in the processed sequence files from mothur (shared and taxonomy) and a design of the sampling. We also load in the environmental data. We then remove the mock community from the dataset and ensure the the design and OTU table are aligned by row.

```
# 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_resgrad.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.pick.opti_m
taxon  <- "data/ul_resgrad.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.pick.opti_m

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

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

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

# Load environmental data
env.dat <- read.csv("data/ResGrad_EnvDat.csv", header = TRUE)
env.dat <- env.dat[-c(16,17,18),]

# Subset to just the reservoir gradient sites
OTUs <- OTUs[str_which(rownames(OTUs), "RG"),]
OTUs <- OTUs[-which(rownames(OTUs) == "RGMockComm"),]

# make sure OTU table matches up with design order
design <- design[-c(34:39),]
OTUs <- OTUs[match(rownames(design), rownames(OTUs)),]
design$distance <- max(na.omit(design$distance)) - design$distance
env.dat$distance <- max(na.omit(env.dat$dist.dam)) - env.dat$dist.dam
```

Clean and transform OTU table

Here, we remove OTUs with low incidence across sites, we remove any samples with low coverage, and we standardize the OTU table by log-transforming the abundances and relativizing by site.

```
# Sequencing Coverage
coverage <- rowSums(OTUs)

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

```r
OTUs <- OTUs[-which(coverage < 10000), ]
design <- design[-which(coverage < 10000), ]
otus.for.inext <-  t(OTUs)
# Remove OTUs with < 2 occurences across all sites
OTUs <- OTUs[, which(colSums(OTUs) >= 2)]
coverage <- rowSums(OTUs)

# Rarify the community, nest RNA in DNA, and reorganize OTU table
set.seed(47405)
OTUs <- rrarefy(OTUs, min(coverage))
OTUs.w.dna <- OTUs[which(design$type == "water" & design$molecule == "DNA"),]
rowSums((OTUs.w.dna > 1))
```

```
## RGD01 RGD02 RGD03 RGD04 RGD06 RGD07 RGD08 RGD09 RGD10 RGD11 RGD12 RGD13
##   319   405   468   372   415   693   545   704   687  1050  1387   515
## RGD14 RGD15
##   548  1313
```

```r
OTUs.w.rna <- OTUs[which(design$type == "water" & design$molecule == "RNA"),]
rowSums((OTUs.w.rna > 1))
```

```
## RGc01 RGc02 RGc03 RGc04 RGc05 RGc07 RGc08 RGc09 RGc10 RGc11 RGc12 RGc13
##   130   142    69   283   142    56   101   162   462   159   185   163
## RGc14 RGc15
##   108   107
```

```r
OTUs.w.dna <- OTUs.w.dna + as.matrix(decostand(OTUs.w.rna, method = "pa"))
rowSums((OTUs.w.dna > 1))
```

```
## RGD01 RGD02 RGD03 RGD04 RGD06 RGD07 RGD08 RGD09 RGD10 RGD11 RGD12 RGD13
##   325   412   472   385   429   699   554   712   741  1065  1396   531
## RGD14 RGD15
##   566  1321
```

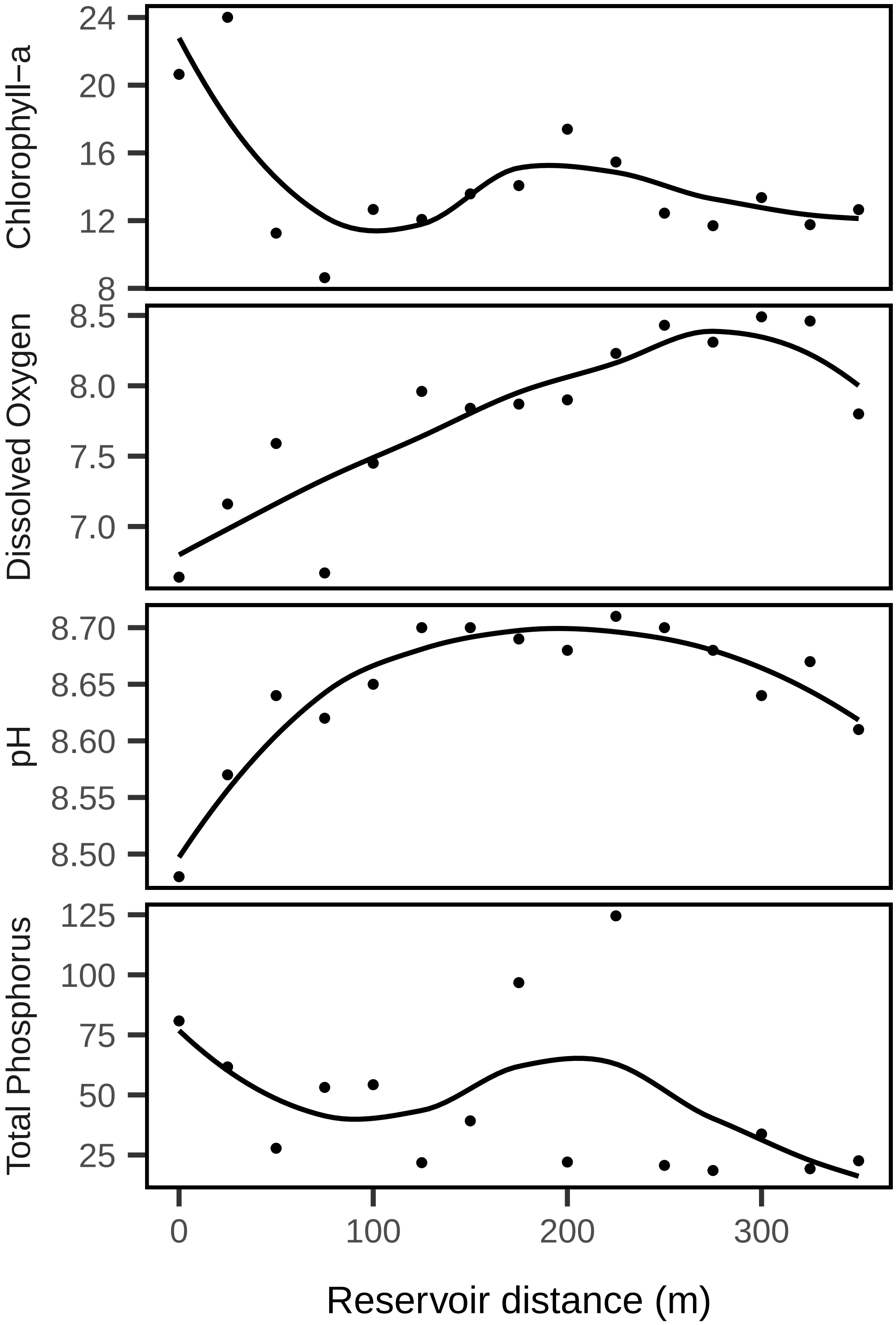
```r
OTUs <- rbind(OTUs[1:3,],
              OTUs.w.dna,
              OTUs.w.rna)
OTUs <- OTUs[match(rownames(design), rownames(OTUs)),]

# Make Relative Abundance Matrices
OTUsREL <- decostand(OTUs, method = "total")

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

Figure S1: Reservoir environmental gradients

Just to see if there are any strong underlying resource or nutrient gradients in the reservoir, we'll plot them along the distance of the reservoir.

```r
facet.labs <- c(`chla` = "Chlorophyll-a",
                `color` = "Color",
                `DO` = "Dissolved Oxygen",
                `pH` = "pH",
                `TP` = "Total Phosphorus")
```

```
env.dat %>% select(distance, DO, pH, TP, chla) %>%
  gather(variable, value, -distance) %>%
  ggplot(aes(x = distance, y = value)) +
  geom_point() +
  geom_smooth(method = "loess", color = "black", se = F) +
  facet_grid(variable ~., scales = "free", switch = "y",
             labeller = as_labeller(facet.labs)) +
  theme(strip.background = element_blank(),
        strip.text = element_text(size = 14),
        strip.placement = "outside") +
  labs(x = "Reservoir distance (m)",
       y = "") +
  scale_y_continuous() +
  ggsave("figures/FigureS1.pdf", height = 3/4*4*3, width = 4, units = "in")
```



So, there are some weak gradients, but nothing too prevailing.

Analyze Diversity

Now, we will analyze the bacterial diversity in the reservoir and nearby soils to figure out how well they support different mechanisms of community assembly.

How does α-diversity vary along the reservoir?

First, we use the method of rarefaction and extrapolation developed by Chao et al. in the iNEXT package. Note: this version of the code loads data from the intermediate-data folder.

```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)
shan <- diversity(OTUs, index = "shannon")
exp.shan <- exp(shan)
alpha.div <- cbind(design, S.obs, simpsE, shan, exp.shan)

# define singleton estimator from Chiu and Chao 2016 PeerJ
source("bin/Chao_functions.R")

# # define function to extract estimated richness
singleton.apply <- function(x){
  singleton.Est(x, "abundance")$corrected.data
}

# This code is commented out, but first applies singleton correction
# then the following line runs the estimateD function
# otus.for.inext <- apply(otus.for.inext, MARGIN = 2, singleton.apply)
# divestim <- estimateD(otus.for.inext, datatype = "abundance",
#           base = "size", conf = 0.95)
# saveRDS(divestim, file = "intermediate-data/inext-output.rda")
divestim <- readRDS("intermediate-data/inext-output.rda")
divestim.df <- divestim %>%
 mutate(habitat = str_to_title(design[as.character(site),"type"]))
```

Next, we'll extract the estimates for the Hill numbers at different levels of q, which differentially weight common versus rare species.

```r
hill.water <- divestim.df %>%
  filter(site %in% rownames(OTUs)) %>%
  left_join(rownames_to_column(alpha.div, var = "site")) %>%
  filter(habitat == "Water")
```

```
## Warning: Column `site` joining factor and character vector, coercing into
## character vector
```

```r
hill.water.rich <- subset(hill.water, order == 0)
hill.water.shan <- subset(hill.water, order == 1)
hill.water.simp <- subset(hill.water, order == 2)

hill.water.mod.rich <- lm(qD ~ distance * molecule, data = hill.water.rich)
hill.water.mod.shan <- lm(qD ~ distance * molecule, data = hill.water.shan)
hill.water.mod.simp <- lm(qD ~ distance * molecule, data = hill.water.simp)

# tidy up the model output
hill.water.mods <- as_tibble(rbind.data.frame(
  tidy(hill.water.mod.rich) %>% add_column(Diversity = "Richness"),
  tidy(hill.water.mod.shan) %>% add_column(Diversity = "Shannon"),
  tidy(hill.water.mod.simp) %>% add_column(Diversity = "Simpson")
))
```

```
# Summary table of the model results.
hill.water.mods %>%
  group_by(Diversity) %>%
  rename("Term" = term,
         "Estimate" = estimate,
         "Std. Error" = std.error,
         "Statistic" = statistic,
         "p-value" = p.value) %>%
  select(Diversity, everything()) %>%
  pander(round = 4)
```

Diversity	Term	Estimate	Std. Error	Statistic	p-value
Richness	(Intercept)	1497	100.6	14.88	0
Richness	distance	-3.176	0.4976	-6.381	0
Richness	moleculeRNA	-1170	142.3	-8.222	0
Richness	distance:moleculeRNA	2.985	0.7003	4.263	3e-04
Shannon	(Intercept)	153.7	19.41	7.921	0
Shannon	distance	-0.2941	0.096	-3.062	0.0053
Shannon	moleculeRNA	-123.9	27.46	-4.513	1e-04
Shannon	distance:moleculeRNA	0.2457	0.1352	1.818	0.0815
Simpson	(Intercept)	55.44	6.47	8.57	0
Simpson	distance	-0.0783	0.032	-2.446	0.0221
Simpson	moleculeRNA	-36.78	9.151	-4.019	5e-04
Simpson	distance:moleculeRNA	0.0402	0.045	0.8918	0.3813

Figure 2: diversity patterns along the gradient

Panel a: alpha diversity

First, generate panel a for Figure 2.

```
# postitions for labels
xpos = max((na.omit(hill.water$distance)))
yposDNA = predict(hill.water.mod.rich, newdata = data.frame(distance = 0, molecule = "DNA"))
yposRNA = predict(hill.water.mod.rich, newdata = data.frame(distance = 0, molecule = "RNA"))

# Here we generate panel a for Figure 2
alpha.fig <- hill.water %>% filter(type == "water", order == 0) %>%
  mutate(molecule = ifelse(molecule == "DNA", "Total", "Active")) %>%
  ggplot(aes(x = distance, y = qD,
             ymin = qD.LCL, ymax = qD.UCL,
             shape = molecule)) +
  # geom_errorbar(size = .5, width = 10, alpha = 0.5) +
  geom_smooth(method = "lm", aes(linetype = molecule), color = "black") +
  geom_point(size =3, alpha = 0.8) +
  labs(x = "Reservoir distance (m)",
       y = "Estimated richness") +
  scale_y_continuous(breaks = seq(0, 2000, by = 500)) +
  scale_x_continuous(limits = c(-49, 350)) +
  theme(legend.position = "none") +
  guides(fill = guide_legend(override.aes=list(fill=NA))) +
  annotate("text", x = -33, y = yposRNA ,
           label = "Active", size = 5) +
  annotate("text", x = -33, y = yposDNA ,
           label = "Total", size = 5) +
  annotate(geom = "text", x = xpos, y = 2000, hjust = 1, vjust = 1, size = 5,
           label = paste0("r^2== ",round(summary(hill.water.mod.rich)$r.squared, 2)), parse = T)
```

Similarity To Terrestrial Habitat Across Gradient (Terrestrial Influence)

Here, we fit a linear model to the similarity of the aquatic community to the soil community.

```
# 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(bray.mean ~ distance * molecule, data = UL.sim)
predict(model.terr, newdata = data.frame(distance = 0, molecule = c("RNA", "DNA")))
```

```
##          1          2
## 0.03090104 0.17193131
```

```
pander(model.terr)
```

Table 2: Fitting linear model: bray.mean ~ distance * molecule

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.1719	0.0138	12.46	5.707e-12
distance	-0.0003988	6.827e-05	-5.841	5.045e-06
moleculeRNA	-0.141	0.01952	-7.226	1.821e-07
distance:moleculeRNA	0.0003839	9.608e-05	3.996	0.0005324

Panel b: beta-diversity

```
ypred.act <- predict(model.terr, newdata = data.frame(distance = 0, molecule = "RNA"))
ypred.tot <- predict(model.terr, newdata = data.frame(distance = 0, molecule = "DNA"))

# make plot
similarity.plot <- UL.sim %>%
  mutate(molecule = ifelse(UL.sim$molecule == "DNA", "Total", "Active")) %>%
  ggplot(aes(x = distance, y = bray.mean, shape = molecule)) +
  geom_smooth(method = "lm", aes(linetype = molecule), color = "black", show.legend = T) +
  geom_point(alpha = 0.8, size = 3, show.legend = T) +
  labs(y = str_wrap("Percent similarity to soil community", width = 20),
       x = "Reservoir distance (m)") +
  theme(legend.position = "none") +
  scale_x_continuous(limits = c(-49,350)) +
  annotate(geom = "text", x = 350, y = max(UL.sim$bray.mean), hjust = 1, vjust = 1, size = 5,
           label = paste0("r^2== ",round(summary(model.terr)$r.squared, 2)), parse = T) +
  annotate("text", x = -33, y = ypred.act, label = "Active", size = 5) +
  annotate("text", x = -33, y = ypred.tot, label = "Total", size = 5)
```

Create combined figure

```
plot_grid(alpha.fig + labs(x = ""), similarity.plot,
          align = "hv",
          labels = "auto", ncol = 1) +
  ggsave("figures/Figure2.pdf")
```

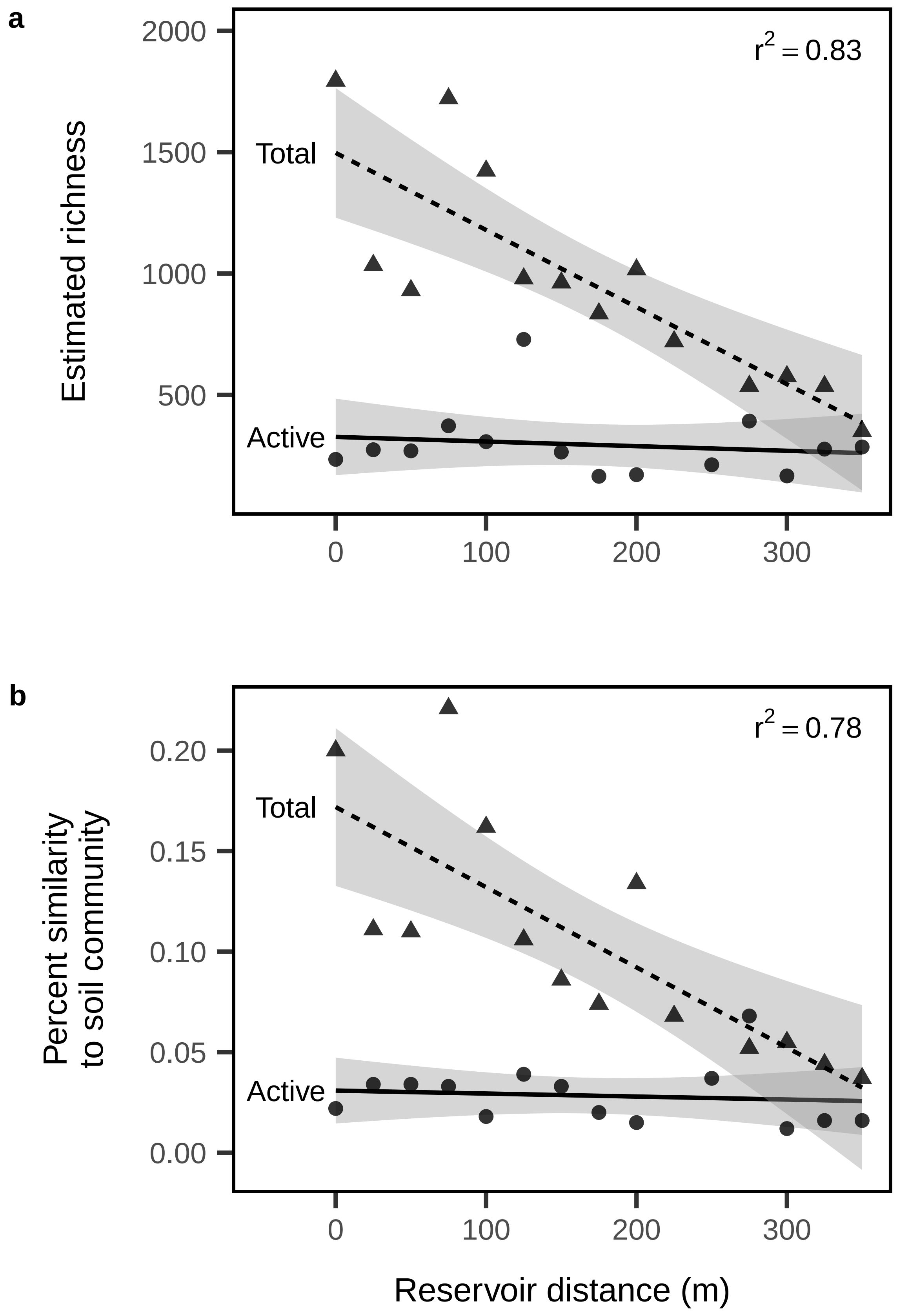
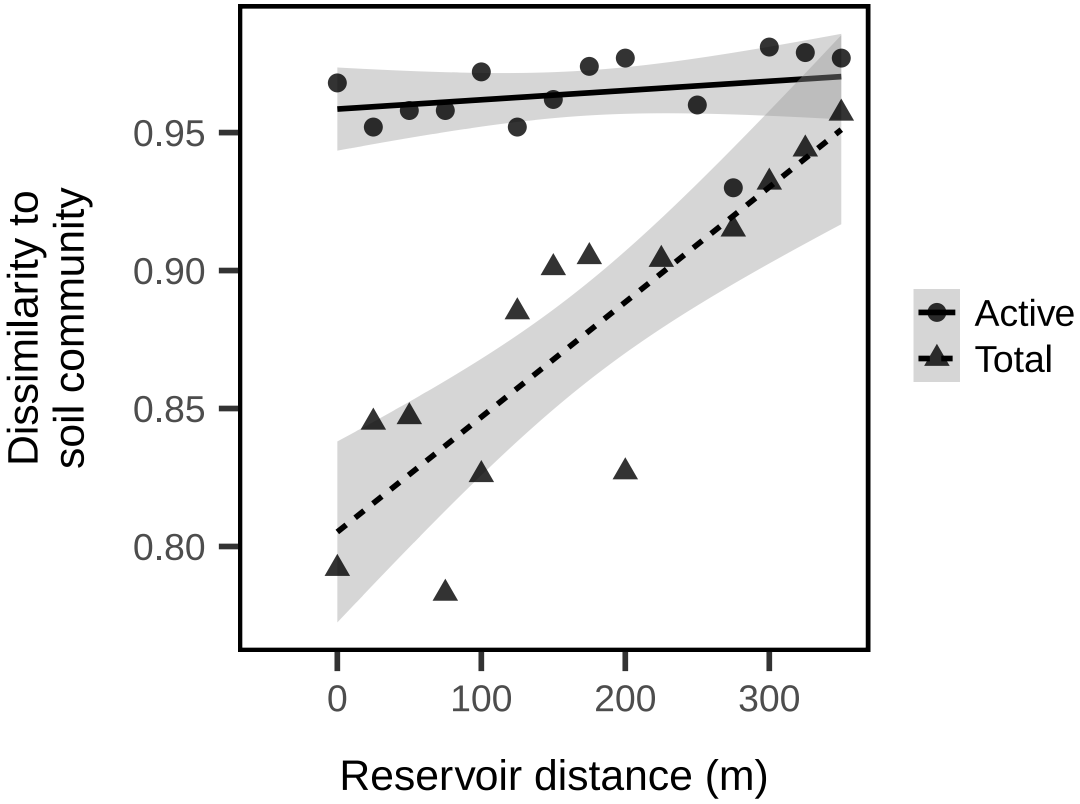
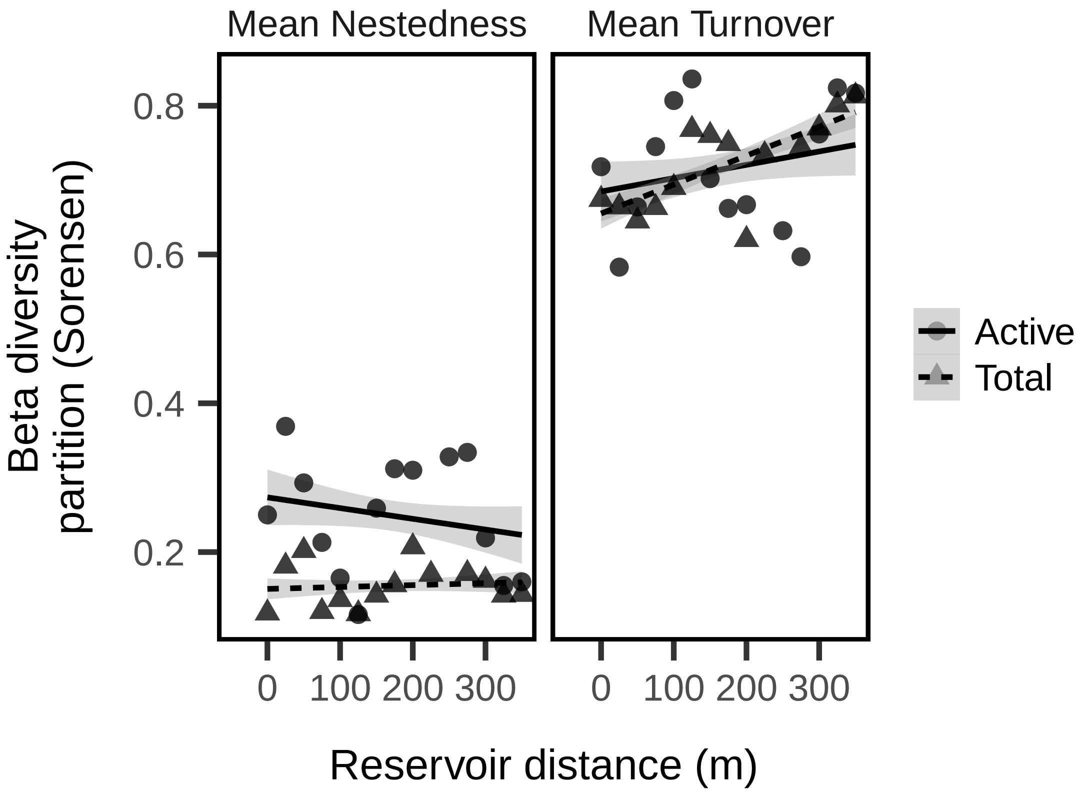

a
Estimated richness
2000
1500
1000
500
r^2 = 0.83
Total
Active
0
100
200
300
b
Percent similarity
to soil community
0.20
0.15
0.10
0.05
0.00
r^2 = 0.78
Total
Active
0
100
200
300
Reservoir distance (m)

Figure S3: Are the aquatic samples nested subsets of the soil?

```
betapart.sor <- beta.pair(decostand(OTUs, method = "pa"), "sorensen")

nest.lake <- as.matrix(betapart.sor$beta.sne)[-c(1:3), 1:3]
nest.mean    <- round(apply(nest.lake, 1, mean), 3)
nest.se      <- round(apply(nest.lake, 1, se), 3)
UL.nest       <- cbind(design[-c(1:3), ], nest.mean, nest.se)

turn.lake <- as.matrix(betapart.sor$beta.sim)[-c(1:3), 1:3]
turn.mean    <- round(apply(turn.lake, 1, mean), 3)
turn.se      <- round(apply(turn.lake, 1, se), 3)
UL.turn       <- cbind(design[-c(1:3), ], turn.mean, turn.se)

sor.lake <- as.matrix(betapart.sor$beta.sor)[-c(1:3), 1:3]
sor.mean    <- round(apply(sor.lake, 1, mean), 3)
sor.se      <- round(apply(sor.lake, 1, se), 3)
UL.sor       <- cbind(design[-c(1:3), ], sor.mean, sor.se)

left_join(UL.nest, UL.turn) %>% left_join(UL.sor) %>%
  mutate(molecule = ifelse(molecule == "DNA", "Total", "Active")) %>%
  ggplot(aes(x = distance, y = sor.mean, shape = molecule)) +
  geom_smooth(method = "lm", aes(linetype = molecule), color = "black", show.legend = T) +
  geom_point(alpha = 0.8, size = 3, show.legend = T) +
  labs(y = str_wrap("Dissimilarity to soil community", width = 20),
       x = "Reservoir distance (m)") +
  scale_x_continuous(limits = c(-49,350))
```



```r
betadivcomp.sor <- beta.div.comp(mat = OTUsREL.log, coef = "S", quant = FALSE, save.abc = FALSE)

rich.lake <- as.matrix(betadivcomp.sor$rich)[-c(1:3), 1:3]
rich.se      <- round(apply(rich.lake, 1, se), 3)
rich.mean    <- round(apply(rich.lake, 1, mean), 3)
UL.rich       <- cbind(design[-c(1:3), ], rich.mean, rich.se)

repl.lake <- as.matrix(betadivcomp.sor$repl)[-c(1:3), 1:3]
repl.mean    <- round(apply(repl.lake, 1, mean), 3)
repl.se      <- round(apply(repl.lake, 1, se), 3)
UL.repl       <- cbind(design[-c(1:3), ], repl.mean, repl.se)

UL_betapartitions <- left_join(UL.nest, UL.turn) %>% left_join(UL.rich) %>% left_join(UL.repl) %>%
  gather(nest.se, turn.se, rich.se, repl.se, key = "partition", value = "se") %>%
  gather(nest.mean, turn.mean, rich.mean, repl.mean, key = "partition", value = "beta")

UL_betapartitions %>%
  mutate(molecule = ifelse(molecule == "DNA", "Total", "Active")) %>%
  mutate(family = ifelse(partition %in% c("nest.mean", "turn.mean"), "Baselga", "Podani")) %>%
  filter(family == "Baselga") %>%
  mutate(partition = ifelse(partition == "nest.mean", "Mean Nestedness", "Mean Turnover")) %>%
  ggplot(aes(x = distance, y = beta, shape = molecule)) +
  geom_smooth(method = "lm", aes(linetype = molecule), color = "black", show.legend = T) +
  geom_point(alpha = 0.3, size = 3, show.legend = T) +
  #geom_errorbar(aes(ymax = beta + se, ymin = beta - se), width = 10) +
  facet_wrap(.~partition) +
  labs(y = str_wrap("Beta diversity partition (Sorensen)", width = 20),
       x = "Reservoir distance (m)") +
  scale_x_continuous(limits = c(-49,350)) +
  ggsave("figures/FigureS3.pdf", width = 8, height = 4)
```



Identifying the Soil Bacteria

Now, we wish to determine whether soil-derived taxa are driving this pattern, and then ask who these influential soil bacteria are.

To classify soil bacteria, we take an incidence-based approach and classify OTUs as:
- present in the soil and present, but never active, in the reservoir
- present in the soil and active in the reservoir

```
# separate lake and soil samples
lake.total <- OTUs[which(design$molecule == "DNA", design$type == "water"),]
soil.total <- OTUs[which(design$molecule == "DNA", design$type == "soil"),]

# which otus are present in both lake and soil samples
lake.and.soil.total <- OTUs[which(design$molecule == "DNA", design$type == "water"),
                            which(colSums(lake.total) > 0 & colSums(soil.total) > 0)]

# isolate just the dna and rna lake communities
w.dna <- OTUs[which(design$molecule == "DNA" & design$type == "water"), ]
w.rna <- OTUs[which(design$molecule == "RNA" & design$type == "water"), ]

# pull out the lake rna counts for otus found in lake and soil
lake.and.soil.act <- w.rna[,colnames(lake.and.soil.total)]

# of these lake and soil taxa, which are never active? active?
nvr.act <- which(colSums(lake.and.soil.act) == 0)
yes.act <- which(colSums(lake.and.soil.act) != 0)
```

```r
# how many otus are active relative to the total number of otus
length(nvr.act) / ncol(lake.and.soil.total) # 88% of soil-derived bac never active
```

```
## [1] 0.8210454
```

```r
length(yes.act) / ncol(soil.total) # 8% of all soil taxa were active in lake
```

```
## [1] 0.1327096
```

```r
# of taxa who were never active, what fraction of the total community did they represent?
sum(rowSums(w.dna[,names(nvr.act)]))
```

```
## [1] 23585
```

```r
sum(rowSums(w.dna[,names(yes.act)]))
```

```
## [1] 499388
```

```r
sum(rowSums(w.dna[,names(nvr.act)])) / sum(rowSums(w.dna))
```

```
## [1] 0.04509793
```

```r
# of taxa who became active, what fraction of the dna community did they represent?
sum(rowSums(w.dna[,names(yes.act)])) / sum(rowSums(w.dna))
```

```
## [1] 0.9549021
```

```r
prop.nvr.act <- rowSums(w.dna[,nvr.act]) / rowSums(w.dna)
# cbind.data.frame(design.dna, inactive = prop.nvr.act) %>%
#   ggplot(aes(x = distance, y = inactive)) +
#   geom_point() +
#   geom_line(stat = "smooth", method = "lm", formula = y ~ x, se = F) +
#   labs(x = "Reservoir transect (m)", y = "Rel. abundance of taxa\n that are never active") +
#   scale_x_reverse()
```

We calculate the richness of the soil taxa that are never active in the lake. We calculate richness from the DNA-based samples.

```r
# pull out their dna abundances and calculate richness
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)
summary(terr.mod1)
```

```
##
## Call:
## lm(formula = terr.rich.log ~ design.dna$distance)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.199417 -0.123300 -0.000783  0.080926  0.234711
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.0266909  0.0726577  41.657 2.37e-14 ***
## design.dna$distance -0.0025661  0.0003595  -7.138 1.18e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1478 on 12 degrees of freedom
## Multiple R-squared:  0.8094, Adjusted R-squared:  0.7935
## F-statistic: 50.95 on 1 and 12 DF,  p-value: 1.184e-05
```

```
T1.R2 <- round(summary(terr.mod1)$r.squared, 2)
T1.int <- terr.mod1$coefficients[1]
T1.slp <- terr.mod1$coefficients[2]
pander(terr.mod1)
```

Table 3: Fitting linear model: terr.rich.log ~ design.dna$distance We find distance is a highly significant predictor of the richness of these soil-derived taxa (on a log-scale).

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	3.027	0.07266	41.66	2.374e-14
design.dna$distance	-0.002566	0.0003595	-7.138	1.184e-05

Figure 3: Fate of terrestrial bacteria

Panel 3a: transients

```
transient.plot <- tibble(transient_rich = terr.rich, distance = design.dna$distance) %>%
  ggplot(aes(x = distance, y = transient_rich)) +
  geom_smooth(method = "lm", color = "black", fill = "grey") +
  geom_point(size = 3, alpha = .8, color = "black") +
  scale_y_log10() +
  annotation_logticks(sides = "l", size = 1) +
  labs(x = "Reservoir distance (m)",
       y = "Inactive soil taxa in reservoir") +
  annotate("text", x = 350, y = max(terr.rich) + 200, hjust = 1, vjust = 0, size = 5,
           label = paste0("r^2== ",T1.R2), parse = T)
```

What is the fate of soil-derived taxa in the reservoir?

So, we observe that most soil-derived taxa appear to decay once they enter the reservoir. Do any soil-derived taxa persist in the active bacterial community of the reservoir and do they rise to high relative abundances?

```
# identify otus in soil samples and lake samples
in.soil <- OTUs[, which(colSums(OTUs[c(1:3),]) > 0 )]
#in.lake <- OTUs[, which(colSums(OTUs[-c(1:3),]) > 0)]

# isolate just the rna water samples and convert to presence-absence
in.lake.rna <- OTUs[which(design$molecule == "RNA" & design$type == "water"), ]
in.lake.rna.pa <- (in.lake.rna > 0) * 1
```

```r
# define the 'core' taxa as otus present in 50% of samples
in.lake.core <- w.dna[, which((colSums(in.lake.rna.pa) / nrow(in.lake.rna.pa)) >= 0.75)]

# of the core, how many are also in the soil samples?
in.lake.core.from.soils <- in.lake.core[, intersect(colnames(in.lake.core), colnames(in.soil))]

# of the core which are not in the soil samples
in.lake.core.not.soils <- in.lake.core[, setdiff(colnames(in.lake.core), colnames(in.soil))]

# Find the relative abundance of the core taxa and prepare data frame to plot
in.lake.core.from.soils.REL <- in.lake.core.from.soils / rowSums(w.dna)

in.soil.to.plot <- as.data.frame(in.lake.core.from.soils.REL) %>%
  rownames_to_column("sample_ID") %>%
  gather(otu_id, rel_abundance, -sample_ID) %>%
  left_join(rownames_to_column(design.dna, "sample_ID")) %>%
  add_column(found = "soils")

in.lake.core.not.soils.REL <- in.lake.core.not.soils / rowSums(w.dna)

in.lake.to.plot <- as.data.frame(in.lake.core.not.soils.REL) %>%
  rownames_to_column("sample_ID") %>%
  gather(otu_id, rel_abundance, -sample_ID) %>%
  left_join(rownames_to_column(design.dna, "sample_ID")) %>%
  add_column(found = "lake")
```

```r
# model distance effect on rel abundance to get slope and pval
soil.core.mods <- apply(in.lake.core.from.soils.REL, MARGIN = 2,
    FUN = function(x) summary(lm(x ~ design.dna$distance))$coefficients[2,c(1,4)])
rownames(soil.core.mods) <- c("slope", "pval")

# classify otus as significantly increasing or decreasing along reservoir
soil.core.decreasing <- as.data.frame(t(soil.core.mods)) %>%
  rownames_to_column("OTU") %>%
  filter(slope < 0) %>%   # rel abund decreases toward dam
  left_join(OTU.tax)
```

```
## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector
```

```r
soil.core.increasing <- as.data.frame(t(soil.core.mods)) %>%
  rownames_to_column("OTU") %>%
  filter(slope > 0) %>%   # rel abund increases toward dam
  left_join(OTU.tax)
```

```
## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector
```

```r
nonsoil.core.mods <- apply(in.lake.core.not.soils.REL, MARGIN = 2,
    FUN = function(x) summary(lm(x ~ design.dna$distance))$coefficients[2,c(1,4)])
rownames(nonsoil.core.mods) <- c("slope", "pval")
nonsoil.core.decreasing <- as.data.frame(t(nonsoil.core.mods)) %>%
  rownames_to_column("OTU") %>%
  filter(slope < 0) %>%   # rel abund decreases toward dam
  left_join(OTU.tax)
```

```
## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector
```

```
nonsoil.core.increasing <- as.data.frame(t(nonsoil.core.mods)) %>%
  rownames_to_column("OTU") %>%
  filter(slope > 0) %>%   # rel abund increases toward dam
  left_join(OTU.tax)
```

```
## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector
```

Table S1 and S2

```
pander(soil.core.decreasing, caption = "Core taxa found in soils that get rarer along the transect.")
```

Table 4: Core taxa found in soils that get rarer along the transect. (continued below)

OTU	slope	pval	Domain	Phylum
Otu00009	-5.115e-05	0.02741	Bacteria	Proteobacteria
Otu00010	-4.281e-05	0.5552	Bacteria	Proteobacteria
Otu00011	-1.928e-05	0.6028	Bacteria	Proteobacteria
Otu00018	-4.637e-05	0.02104	Bacteria	Proteobacteria
Otu00022	-2.499e-05	0.1178	Bacteria	Verrucomicrobia
Otu00028	-3.043e-05	0.02348	Bacteria	Proteobacteria
Otu00030	-2.222e-06	0.2752	Bacteria	Actinobacteria
Otu00039	-8.511e-06	0.1793	Bacteria	Proteobacteria
Otu00045	-7.99e-06	0.5274	Bacteria	Proteobacteria
Otu00059	-6.488e-05	0.02525	Bacteria	Actinobacteria
Otu00065	-5.535e-05	0.02097	Bacteria	Bacteroidetes
Otu00072	-1.884e-05	0.09145	Bacteria	Proteobacteria
Otu00077	-5.843e-05	0.0117	Bacteria	Bacteroidetes
Otu00086	-1.26e-05	0.0353	Bacteria	Proteobacteria
Otu00094	-2.214e-05	0.03137	Bacteria	Proteobacteria
Otu00095	-3.555e-05	0.03573	Bacteria	Proteobacteria
Otu00170	-2.475e-05	0.02842	Bacteria	Bacteroidetes
Otu00545	-1.25e-06	0.0273	Bacteria	Actinobacteria

Table 5: Table continues below

Class	Order
Gammaproteobacteria	Pseudomonadales
Proteobacteria_unclassified	Proteobacteria_unclassified
Betaproteobacteria	Betaproteobacteria_unclassified
Gammaproteobacteria	Pseudomonadales
Opitutae	Opitutae_unclassified
Gammaproteobacteria	Pseudomonadales
Actinobacteria	Actinomycetales
Betaproteobacteria	Burkholderiales
Betaproteobacteria	Burkholderiales
Actinobacteria	Actinomycetales
Sphingobacteriia	Sphingobacteriales

Class	Order
Alphaproteobacteria	Sphingomonadales
Flavobacteriia	Flavobacteriales
Alphaproteobacteria	Rhizobiales
Betaproteobacteria	Burkholderiales
Betaproteobacteria	Burkholderiales
Sphingobacteriia	Sphingobacteriales
Actinobacteria	Solirubrobacterales

Family	Genus
Pseudomonadaceae	Pseudomonas
Proteobacteria_unclassified	Proteobacteria_unclassified
Betaproteobacteria_unclassified	Betaproteobacteria_unclassified
Pseudomonadaceae	Pseudomonas
Opitutae_unclassified	Opitutae_unclassified
Pseudomonadaceae	Pseudomonas
Micrococcaceae	Micrococcus
Comamonadaceae	Comamonas
Oxalobacteraceae	Oxalobacteraceae_unclassified
Micrococcaceae	Arthrobacter
Sphingobacteriaceae	Pedobacter
Sphingomonadaceae	Sphingomonas
Flavobacteriaceae	Flavobacterium
Bradyrhizobiaceae	Bradyrhizobium
Oxalobacteraceae	Duganella
Comamonadaceae	Comamonadaceae_unclassified
Sphingobacteriaceae	Sphingobacteriaceae_unclassified
Solirubrobacteraceae	Solirubrobacter

```r
pander(soil.core.increasing, caption = "Core taxa found in soils that get more common along the transec
```

Table 7: Core taxa found in soils that get more common along the transect. (continued below)

OTU	slope	pval	Domain	Phylum
Otu00001	1.437e-05	0.07357	Bacteria	Proteobacteria
Otu00002	0.0002104	0.002241	Bacteria	Actinobacteria
Otu00003	9.845e-05	0.006345	Bacteria	Verrucomicrobia
Otu00005	3.593e-05	0.01749	Bacteria	Bacteroidetes
Otu00006	6.515e-06	0.1629	Bacteria	Bacteroidetes
Otu00012	7.565e-06	0.09337	Bacteria	Proteobacteria
Otu00014	8.415e-05	0.0007944	Bacteria	Actinobacteria
Otu00023	3.479e-07	0.7837	Bacteria	Proteobacteria
Otu00029	3.301e-05	0.004547	Bacteria	Actinobacteria
Otu00032	3.59e-06	0.8316	Bacteria	Bacteroidetes
Otu00033	9.093e-06	0.7077	Bacteria	Proteobacteria

Table 8: Table continues below

Class	Order	Family	Genus
Betaproteobacteria	Burkholderiales	Comamonadaceae	Comamonadaceae_unclassified
Actinobacteria	Actinomycetales	Actinomycetales_unclassified	Actinomycetales_unclassified
Spartobacteria	Spartobacteria_unclassified	Spartobacteria_unclassified	Spartobacteria_unclassified
Sphingobacteriia	Sphingobacteriales	Chitinophagaceae	Sediminibacterium
Sphingobacteriia	Sphingobacteriales	Saprospiraceae	Saprospiraceae_unclassified
Betaproteobacteria	Burkholderiales	Comamonadaceae	Comamonadaceae_unclassified
Actinobacteria	Actinomycetales	Actinomycetales_unclassified	Actinomycetales_unclassified
Gammaproteobacteria	Pseudomonadales	Moraxellaceae	Acinetobacter
Actinobacteria	Actinomycetales	Actinomycetales_unclassified	Actinomycetales_unclassified
Bacteroidetes_unclassified	Bacteroidetes_unclassified	Bacteroidetes_unclassified	Bacteroidetes_unclassified
Alphaproteobacteria	Rhizobiales	Rhizobiales_unclassified	Rhizobiales_unclassified

Panel 3b: Trajectories of terrestrial taxa along the gradient

```r
df1 <- as.data.frame(OTUsREL[,nonsoil.core.increasing$OTU]) %>%
  rownames_to_column("sampleID") %>%
  left_join(rownames_to_column(design, "sampleID")) %>%
  gather(OTU, rel_abund, -station, -molecule, -type, -distance, -sampleID) %>%
  filter(molecule == "DNA") %>% left_join(OTU.tax) %>%
  mutate(soils = "Absent from soils", change = "Increasing")
```

```
## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector
```

```r
n1 <- length(unique(df1$OTU))

df2 <- as.data.frame(OTUsREL[,soil.core.increasing$OTU]) %>%
  rownames_to_column("sampleID") %>%
  left_join(rownames_to_column(design, "sampleID")) %>%
  gather(OTU, rel_abund, -station, -molecule, -type, -distance, -sampleID) %>%
  filter(molecule == "DNA") %>% left_join(OTU.tax) %>%
  mutate(soils = "Present in soils", change = "Increasing")
```

```
## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector
```

```r
n2 <- length(unique(df2$OTU))

df3 <- as.data.frame(OTUsREL[,soil.core.decreasing$OTU]) %>%
  rownames_to_column("sampleID") %>%
  left_join(rownames_to_column(design, "sampleID")) %>%
  gather(OTU, rel_abund, -station, -molecule, -type, -distance, -sampleID) %>%
  filter(molecule == "DNA") %>% left_join(OTU.tax) %>%
  mutate(soils = "Present in soils", change = "Decreasing")
```

```
## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector
```

```r
n3 <- length(unique(df3$OTU))

df4 <- as.data.frame(OTUsREL[,nonsoil.core.decreasing$OTU]) %>%
  rownames_to_column("sampleID") %>%
  left_join(rownames_to_column(design, "sampleID")) %>%
  gather(OTU, rel_abund, -station, -molecule, -type, -distance, -sampleID) %>%
  filter(molecule == "DNA") %>% left_join(OTU.tax) %>%
  mutate(soils = "Absent from soils", change = "Decreasing")
```

```
## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector
```

```r
n4 <- length(unique(df4$OTU))


df.plot <- as_tibble(rbind.data.frame(df1, df2, df3, df4)) %>% filter(type == "water")

taxon_fate.plot <- df.plot %>% mutate(rel_abund = ifelse(rel_abund == 0, 1e-6, rel_abund)) %>%
  filter(soils == "Present in soils") %>%
  #mutate(change = ifelse(change == "Increasing",
  #                       paste0("Increasing (n = ", n2,")"),
  #                       paste0("Decreasing (n = ", n3,")"))) %>%
  ggplot(aes(x = distance, y = rel_abund, group = OTU)) +
  #geom_jitter(alpha = 0.15) +
  geom_line(stat = "smooth", alpha = 0.3, size = 1,
            method = "loess", span = .7, se = FALSE) +
  scale_y_log10(labels = scales::scientific) +
  scale_x_continuous(limits = c(0,380)) +
  #theme(legend.position = "none") +
  #guides(color = guide_legend(ncol = 1)) +
  labs(x = "Reservoir distance (m)",
       y = "Active relative abundance") +
  annotate("text", x = 365, y = 1e-1, size = 5, hjust = 1, vjust = 1, angle = 90,
           label = "Maintained") +
  annotate("text", x = 365, y = 1e-5, size = 5, hjust = 0.5, vjust = 1, angle = 90,
           label = "Decaying")

# how much do the different core components contribute to total abundances
in.lake.core.soil.REL <- rowSums(in.lake.core.from.soils) / rowSums(w.dna)
in.lake.core.water.REL <- rowSums(in.lake.core.not.soils) / rowSums(w.dna)
```

Figure 3

```r
plot_grid(transient.plot + labs(x = ""),
          taxon_fate.plot,
          align = "hv", axis = "rltb",
          labels = "auto",
          ncol = 1) +
  ggsave("figures/Figure3.pdf")
```

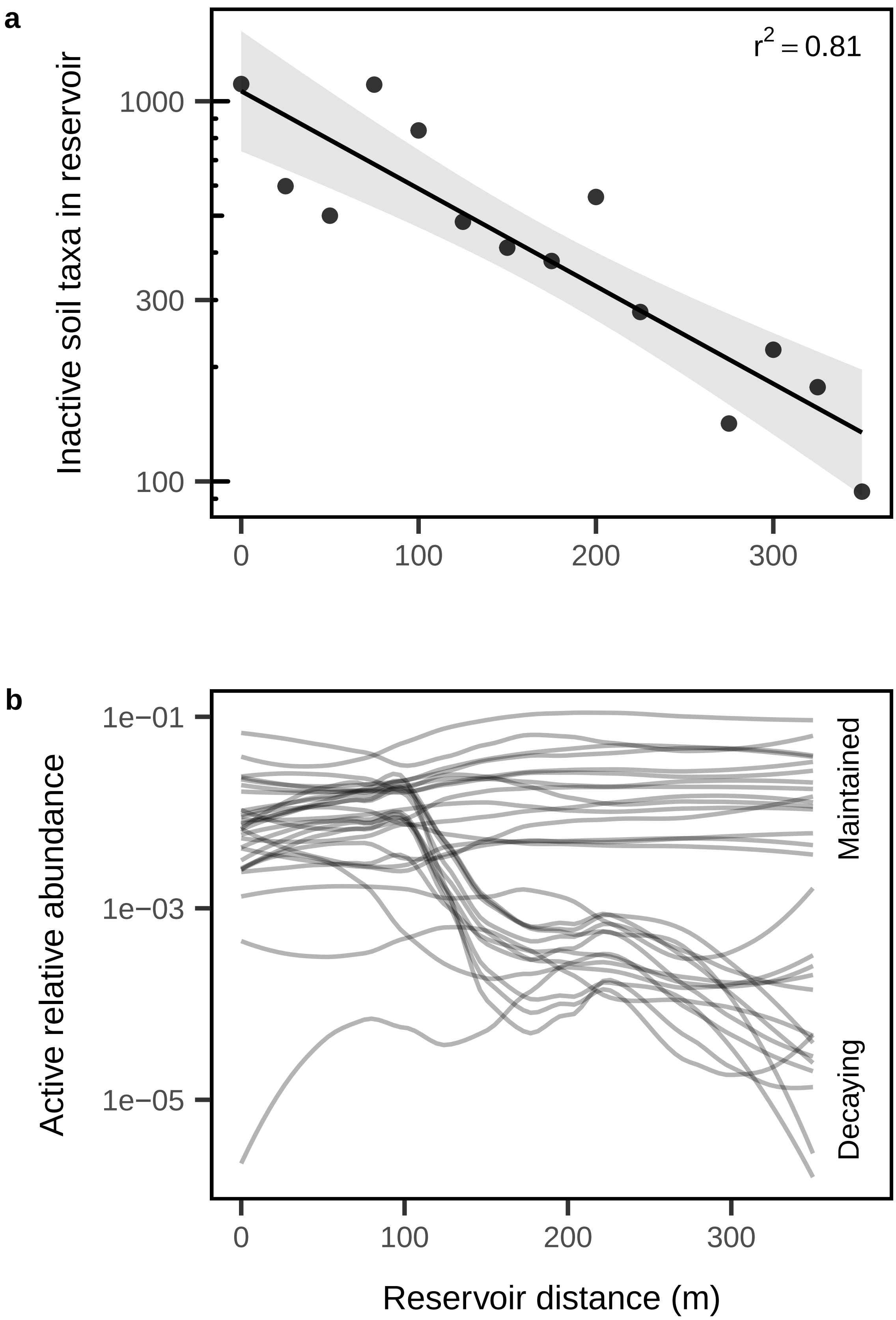

a
r² = 0.81
Inactive soil taxa in reservoir
1000
300
100
0
100
200
300
b
1e−01
1e−03
1e−05
Active relative abundance
Maintained
Decaying
0
100
200
300
Reservoir distance (m)

Figure S4: See which taxa are shared between habitats

```
OTUs.PA <- decostand(OTUsREL, method = "pa")
soil <- names(which(colSums(OTUs.PA[design$type == "soil",]) > 0))
water.dna <- names(which(colSums(OTUs.PA[design$type == "water" & design$molecule == "DNA",]) > 0))
water.rna <- names(which(colSums(OTUs.PA[design$type == "water" & design$molecule == "RNA",]) > 0))

sum(water.rna %in% water.dna)
```

```
## [1] 2085
```

```
nsoil <- length(soil)
nwdna <- length(water.dna)
nwrna <- length(water.rna)
otus.by.habitat <- list("Soil" = soil, "Total Aquatic" = water.dna, "Active Aquatic" = water.rna)

venn.diagram(otus.by.habitat, "figures/FigureS4.png",
             imagetype = "png",
             fontfamily = "sans",
             cat.fontfamily = "sans",
             alpha = .25)
```

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

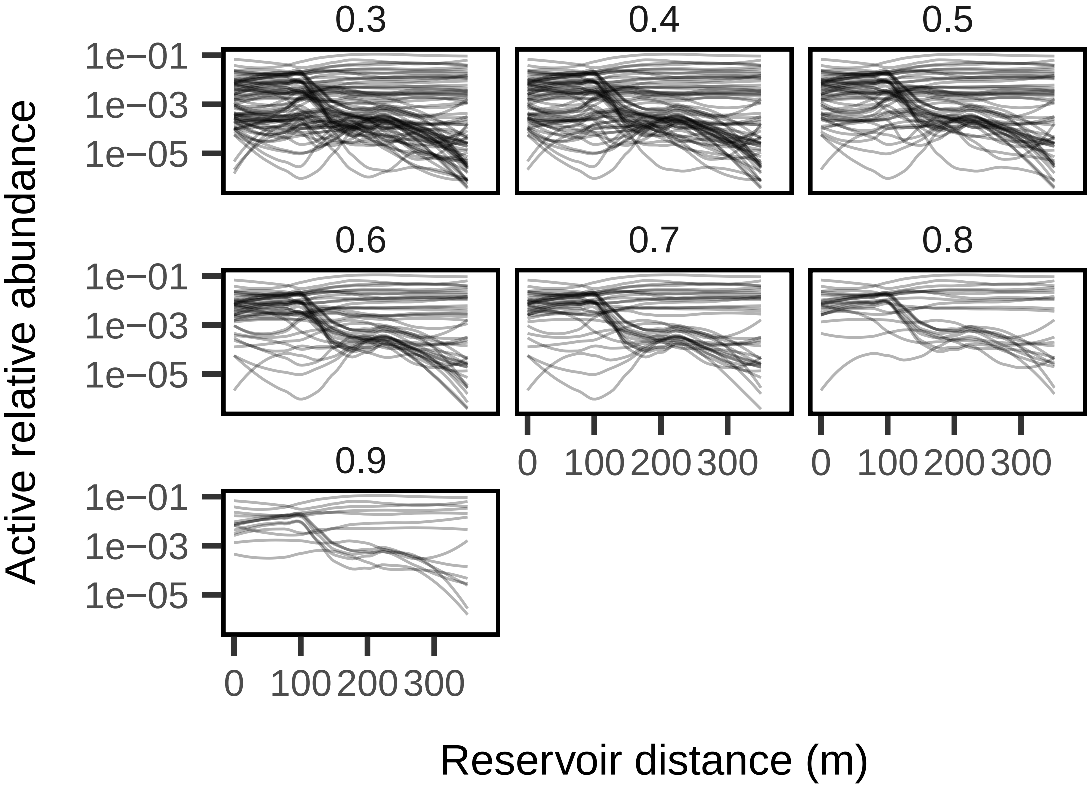
Figure S2: Threshold for cutoffs in occupancy fraction

```
# identify otus in soil samples and lake samples
in.soil <- OTUs[, which(colSums(OTUs[c(1:3),]) > 0 )]

# isolate just the rna water samples and convert to presence-absence
in.lake.rna <- OTUs[which(design$molecule == "RNA" & design$type == "water"), ]
in.lake.rna.pa <- (in.lake.rna > 0) * 1

threshlist <- c(.3, .4, .5, .6, .7, .8, .9)
df.plot <- data.frame()
for(thresh in threshlist){
  # define the 'core' taxa as otus present in 50% of samples
in.lake.core <- w.dna[, which((colSums(in.lake.rna.pa) / nrow(in.lake.rna.pa)) >= thresh)]

# of the core, how many are also in the soil samples?
in.lake.core.from.soils <- in.lake.core[, intersect(colnames(in.lake.core), colnames(in.soil))]

# of the core which are not in the soil samples
in.lake.core.not.soils <- in.lake.core[, setdiff(colnames(in.lake.core), colnames(in.soil))]

# Find the relative abundance of the core taxa and prepare data frame to plot
in.lake.core.from.soils.REL <- in.lake.core.from.soils / rowSums(w.dna)

in.soil.to.plot <- as.data.frame(in.lake.core.from.soils.REL) %>%
  rownames_to_column("sample_ID") %>%
  gather(otu_id, rel_abundance, -sample_ID) %>%
  left_join(rownames_to_column(design.dna, "sample_ID")) %>%
  add_column(found = "soils")

in.lake.core.not.soils.REL <- in.lake.core.not.soils / rowSums(w.dna)

in.lake.to.plot <- as.data.frame(in.lake.core.not.soils.REL) %>%
  rownames_to_column("sample_ID") %>%
  gather(otu_id, rel_abundance, -sample_ID) %>%
  left_join(rownames_to_column(design.dna, "sample_ID")) %>%
  add_column(found = "lake")

# model distance effect on rel abundance to get slope and pval
soil.core.mods <- apply(in.lake.core.from.soils.REL, MARGIN = 2,
    FUN = function(x) summary(lm(x ~ design.dna$distance))$coefficients[2,c(1,4)])
rownames(soil.core.mods) <- c("slope", "pval")

# classify otus as significantly increasing or decreasing along reservoir
soil.core.decreasing <- as.data.frame(t(soil.core.mods)) %>%
  rownames_to_column("OTU") %>%
  filter(slope < 0) %>%   # rel abund decreases toward dam
  left_join(OTU.tax)
soil.core.increasing <- as.data.frame(t(soil.core.mods)) %>%
  rownames_to_column("OTU") %>%
  filter(slope > 0) %>%   # rel abund increases toward dam
  left_join(OTU.tax)

nonsoil.core.mods <- apply(in.lake.core.not.soils.REL, MARGIN = 2,
    FUN = function(x) summary(lm(x ~ design.dna$distance))$coefficients[2,c(1,4)])
rownames(nonsoil.core.mods) <- c("slope", "pval")
nonsoil.core.decreasing <- as.data.frame(t(nonsoil.core.mods)) %>%
  rownames_to_column("OTU") %>%
  filter(slope < 0) %>%   # rel abund decreases toward dam
  left_join(OTU.tax)
nonsoil.core.increasing <- as.data.frame(t(nonsoil.core.mods)) %>%
  rownames_to_column("OTU") %>%
  filter(slope > 0) %>%   # rel abund increases toward dam
  left_join(OTU.tax)

df1 <- as.data.frame(OTUsREL[,nonsoil.core.increasing$OTU]) %>%
  rownames_to_column("sampleID") %>%
  left_join(rownames_to_column(design, "sampleID")) %>%
  gather(OTU, rel_abund, -station, -molecule, -type, -distance, -sampleID) %>%
  filter(molecule == "DNA") %>% left_join(OTU.tax) %>%
  mutate(soils = "Absent from soils", change = "Increasing")
n1 <- length(unique(df1$OTU))

df2 <- as.data.frame(OTUsREL[,soil.core.increasing$OTU]) %>%
  rownames_to_column("sampleID") %>%
  left_join(rownames_to_column(design, "sampleID")) %>%
  gather(OTU, rel_abund, -station, -molecule, -type, -distance, -sampleID) %>%
  filter(molecule == "DNA") %>% left_join(OTU.tax) %>%
  mutate(soils = "Present in soils", change = "Increasing")
n2 <- length(unique(df2$OTU))
```

```
df3 <- as.data.frame(OTUsREL[,soil.core.decreasing$OTU]) %>%
  rownames_to_column("sampleID") %>%
  left_join(rownames_to_column(design, "sampleID")) %>%
  gather(OTU, rel_abund, -station, -molecule, -type, -distance, -sampleID) %>%
  filter(molecule == "DNA") %>% left_join(OTU.tax) %>%
  mutate(soils = "Present in soils", change = "Decreasing")
n3 <- length(unique(df3$OTU))

df4 <- as.data.frame(OTUsREL[,nonsoil.core.decreasing$OTU]) %>%
  rownames_to_column("sampleID") %>%
  left_join(rownames_to_column(design, "sampleID")) %>%
  gather(OTU, rel_abund, -station, -molecule, -type, -distance, -sampleID) %>%
  filter(molecule == "DNA") %>% left_join(OTU.tax) %>%
  mutate(soils = "Absent from soils", change = "Decreasing")
n4 <- length(unique(df4$OTU))


df.plot <- as_tibble(rbind.data.frame(df1, df2, df3, df4)) %>%
  mutate(thresh = thresh) %>% filter(type == "water") %>%
  bind_rows(df.plot)

}
```

```
## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector

## Warning: Column `OTU` joining character vector and factor, coercing into
## character vector
```

```
taxon_fate.plot <- df.plot %>% mutate(rel_abund = ifelse(rel_abund == 0, 1e-6, rel_abund)) %>%
  filter(soils == "Present in soils") %>%
  #mutate(change = ifelse(change == "Increasing",
  #                       paste0("Increasing (n = ", n2,")"),
  #                       paste0("Decreasing (n = ", n3,")"))) %>%
  ggplot(aes(x = distance, y = rel_abund, group = OTU)) +
  #geom_jitter(alpha = 0.15) +
  geom_line(stat = "smooth", alpha = 0.3, size = .5,
            method = "loess", span = .7, se = FALSE) +
  scale_y_log10(labels = scales::scientific) +
  scale_x_continuous(limits = c(0,380)) +
  facet_wrap(~thresh) +
  #theme(legend.position = "none") +
  #guides(color = guide_legend(ncol = 1)) +
  labs(x = "Reservoir distance (m)",
       y = "Active relative abundance") +
  # annotate("text", x = 365, y = 1e-1, size = 5, hjust = 1, vjust = 1, angle = 90,
  #          label = "Maintained") +
  # annotate("text", x = 365, y = 1e-5, size = 5, hjust = 0.5, vjust = 1, angle = 90,
  #          label = "Decaying") +
  ggsave("figures/FigureS2.pdf", width = 8, height = 6, units = "in")
taxon_fate.plot
```


0.3
0.4
0.5
0.6
0.7
0.8
0.9
1e−01
1e−03
1e−05
0
100
200
300
Active relative abundance
Reservoir distance (m)