Dormancy and dispersal structure bacterial communities across ecosystem boundaries

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

30 May, 2019

Initial Setup

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

```
# Import Required Packages
library("png")
library("grid")
library("tidyverse")
library("vegan")
library("xtable")
library("viridis")
library("cowplot")
library("adespatial")
library("ggrepel")
library("gganimate")
library("maps")
library("rgdal")
library("iNEXT")
library("officer")
library("flextable") #must have gdtools installed also
library("broom")
library("ggpmisc")
library("pander")
library("lubridate")

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.

```
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[-16,]

# 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
OTUs <- OTUs[match(rownames(design), rownames(OTUs)),]
```

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.

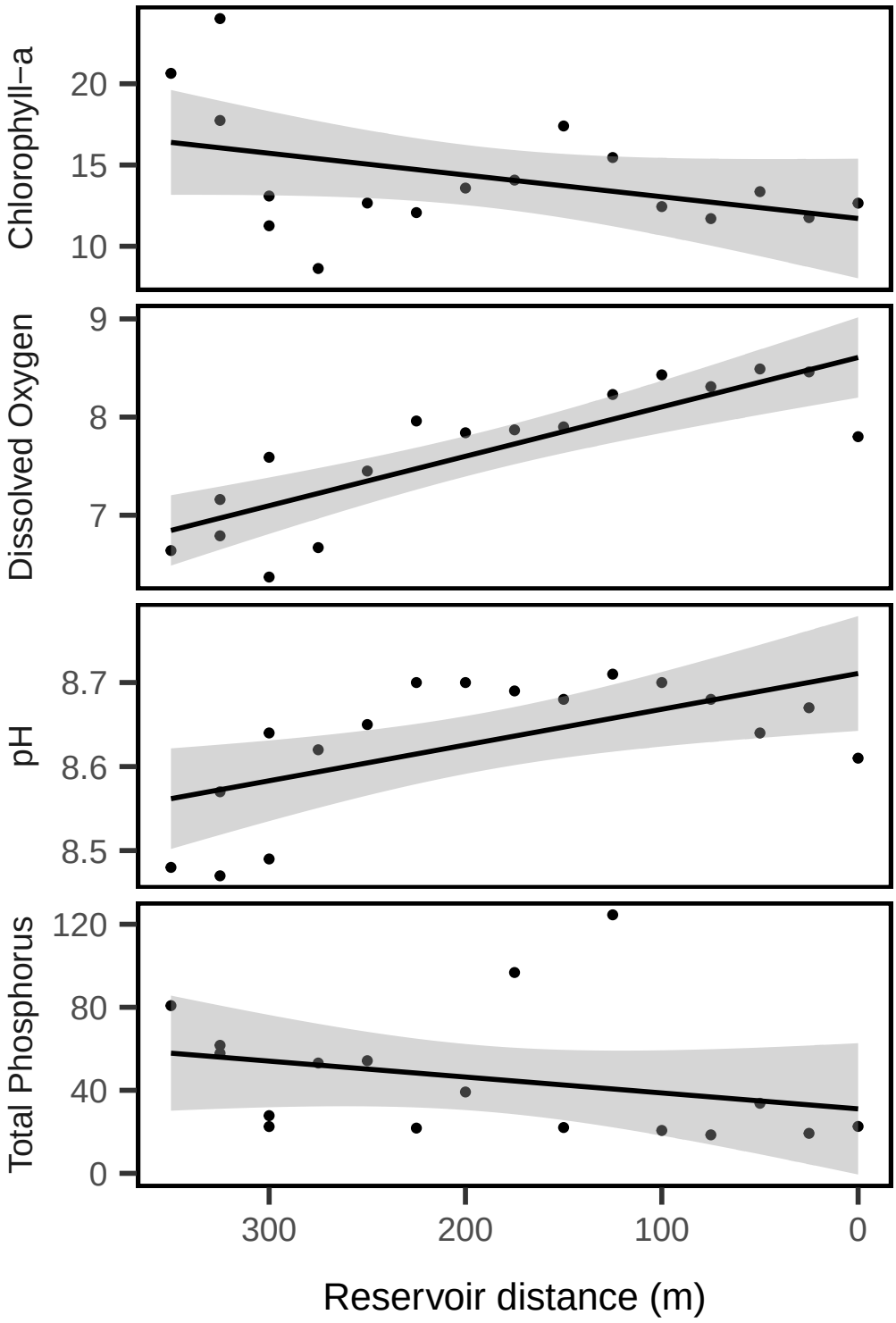
```
# Remove OTUs with less than two occurences across all sites
#OTUs <- OTUs[, which(colSums(OTUs) >= 2)]

# Sequencing Coverage
```

```
coverage <- rowSums(OTUs)

# Remove Low Coverage Samples (This code removes two sites: Site 5DNA, Site 6cDNA)
lows <- which(coverage < 10000)
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)
set.seed(47405)
OTUs <- rrarefy(OTUs, min(coverage))

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

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

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.

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

env.dat %>% select(dist.dam, DO, pH, TP, chla) %>%
  gather(variable, value, -dist.dam) %>%
  ggplot(aes(x = dist.dam, y = value)) +
  geom_point() +
  geom_smooth(method = "lm", color = "black") +
  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_x_reverse() +
  scale_y_continuous()
```



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.

```
# 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")

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

otus.for.inext <- apply(otus.for.inext, MARGIN = 2, singleton.apply)
# divestim <- estimateD(otus.for.inext, datatype = "abundance",
#           base = "size", conf = 0.95)
# divestim <- iNEXT(otus.for.inext, datatype = "abundance",
#                   size =  min(coverage), nboot = 999)
# divestim$iNextEst
# saveRDS(divestim, file = "intermediate-data/inext-output.rda")
divestim <- read_rds("intermediate-data/inext-output.rda")
divestim
```

```
##         site     m       method order    SC       qD   qD.LCL   qD.UCL
## 1   RGSoil01 37027 interpolated     0 0.955 3889.214 3867.259 3911.169
## 2   RGSoil01 37027 interpolated     1 0.955  752.963  746.212  759.715
## 3   RGSoil01 37027 interpolated     2 0.955  182.356  179.896  184.817
## 4   RGSoil02 37027 interpolated     0 0.945 4392.817 4367.251 4418.382
## 5   RGSoil02 37027 interpolated     1 0.945  666.077  660.260  671.895
## 6   RGSoil02 37027 interpolated     2 0.945  134.658  132.742  136.574
## 7   RGSoil03 37027 interpolated     0 0.950 4191.294 4161.602 4220.987
## 8   RGSoil03 37027 interpolated     1 0.950  773.790  766.732  780.848
## 9   RGSoil03 37027 interpolated     2 0.950  179.353  176.344  182.363
## 10     RGD01 37027     observed     0 0.999  350.000  341.907  358.093
## 11     RGD01 37027     observed     1 0.999   60.336   59.421   61.251
## 12     RGD01 37027     observed     2 0.999   30.661   30.111   31.211
## 13     RGc01 37027 interpolated     0 0.996  285.774  277.158  294.391
## 14     RGc01 37027 interpolated     1 0.996   21.184   21.034   21.335
## 15     RGc01 37027 interpolated     2 0.996    9.764    9.671    9.858
## 16     RGD02 37027 interpolated     0 0.996  535.631  521.557  549.704
```

```
## 17     RGD02 37027 interpolated     1 0.996   70.802   69.799   71.805
## 18     RGD02 37027 interpolated     2 0.996   35.622   35.059   36.185
## 19     RGc02 37027 interpolated     0 0.996  276.963  269.657  284.269
## 20     RGc02 37027 interpolated     1 0.996   31.118   30.905   31.332
## 21     RGc02 37027 interpolated     2 0.996   16.072   15.940   16.204
## 22     RGD03 37027 interpolated     0 0.997  576.480  562.427  590.532
## 23     RGD03 37027 interpolated     1 0.997   67.500   66.450   68.550
## 24     RGD03 37027 interpolated     2 0.997   31.838   31.267   32.409
## 25     RGc03 37027 interpolated     0 0.997  166.654  160.265  173.044
## 26     RGc03 37027 interpolated     1 0.997    7.545    7.497    7.593
## 27     RGc03 37027 interpolated     2 0.997    4.194    4.165    4.223
## 28     RGD04 37027 interpolated     0 0.996  536.871  519.687  554.055
## 29     RGD04 37027 interpolated     1 0.996   71.051   70.051   72.052
## 30     RGD04 37027 interpolated     2 0.996   35.457   34.821   36.092
## 31     RGc04 37027 interpolated     0 0.997  392.580  385.414  399.746
## 32     RGc04 37027 interpolated     1 0.997    2.241    2.218    2.264
## 33     RGc04 37027 interpolated     2 0.997    1.336    1.331    1.341
## 34     RGc05 37027 interpolated     0 0.998  212.420  204.739  220.101
## 35     RGc05 37027 interpolated     1 0.998    4.881    4.840    4.923
## 36     RGc05 37027 interpolated     2 0.998    3.967    3.950    3.984
## 37     RGD06 37027 interpolated     0 0.992  720.373  709.705  731.041
## 38     RGD06 37027 interpolated     1 0.992   61.376   60.858   61.894
## 39     RGD06 37027 interpolated     2 0.992   26.153   25.921   26.386
## 40     RGD07 37027 interpolated     0 0.991 1016.407  994.401 1038.413
## 41     RGD07 37027 interpolated     1 0.991   85.475   83.864   87.085
## 42     RGD07 37027 interpolated     2 0.991   34.786   34.100   35.471
## 43     RGc07 37027 interpolated     0 0.997  171.638  163.075  180.202
## 44     RGc07 37027 interpolated     1 0.997    4.496    4.467    4.524
## 45     RGc07 37027 interpolated     2 0.997    3.192    3.172    3.213
## 46     RGD08 37027 interpolated     0 0.992  835.316  824.174  846.458
## 47     RGD08 37027 interpolated     1 0.992   71.572   70.913   72.230
## 48     RGD08 37027 interpolated     2 0.992   29.885   29.555   30.216
## 49     RGc08 37027 interpolated     0 0.998  165.011  160.172  169.850
## 50     RGc08 37027 interpolated     1 0.998   18.257   18.159   18.355
## 51     RGc08 37027 interpolated     2 0.998   10.562   10.482   10.642
## 52     RGD09 37027 interpolated     0 0.993  962.514  942.906  982.123
## 53     RGD09 37027 interpolated     1 0.993  102.957  101.246  104.668
## 54     RGD09 37027 interpolated     2 0.993   40.437   39.617   41.256
## 55     RGc09 37027 interpolated     0 0.997  264.910  257.723  272.096
## 56     RGc09 37027 interpolated     1 0.997    5.931    5.883    5.979
## 57     RGc09 37027 interpolated     2 0.997    3.899    3.879    3.920
## 58     RGD10 37027 interpolated     0 0.992  979.243  968.583  989.904
## 59     RGD10 37027 interpolated     1 0.992  115.134  114.131  116.138
## 60     RGD10 37027 interpolated     2 0.992   50.536   49.946   51.126
## 61     RGc10 37027 interpolated     0 0.993  728.724  712.946  744.503
## 62     RGc10 37027 interpolated     1 0.993   78.838   77.746   79.930
## 63     RGc10 37027 interpolated     2 0.993   29.012   28.430   29.595
## 64     RGD11 37027 interpolated     0 0.990 1423.107 1411.926 1434.289
## 65     RGD11 37027 interpolated     1 0.990  161.982  160.650  163.315
## 66     RGD11 37027 interpolated     2 0.990   65.095   64.432   65.759
## 67     RGc11 37027 interpolated     0 0.996  307.585  299.254  315.916
## 68     RGc11 37027 interpolated     1 0.996   36.292   36.060   36.524
## 69     RGc11 37027 interpolated     2 0.996   22.636   22.483   22.788
## 70     RGD12 37027 interpolated     0 0.991 1720.686 1709.731 1731.640
```

```
## 71     RGD12 37027 interpolated     1 0.991  252.525  250.280  254.770
## 72     RGD12 37027 interpolated     2 0.991   85.267   84.458   86.077
## 73     RGc12 37027 interpolated     0 0.995  372.791  363.552  382.029
## 74     RGc12 37027 interpolated     1 0.995   24.840   24.682   24.997
## 75     RGc12 37027 interpolated     2 0.995   17.702   17.624   17.780
## 76     RGD13 37027 interpolated     0 0.988  930.870  916.712  945.028
## 77     RGD13 37027 interpolated     1 0.988   56.414   55.942   56.885
## 78     RGD13 37027 interpolated     2 0.988   23.056   22.824   23.287
## 79     RGc13 37027 interpolated     0 0.997  269.903  263.231  276.575
## 80     RGc13 37027 interpolated     1 0.997   15.722   15.619   15.825
## 81     RGc13 37027 interpolated     2 0.997   10.745   10.689   10.800
## 82     RGD14 37027 interpolated     0 0.986 1034.420 1017.730 1051.109
## 83     RGD14 37027 interpolated     1 0.986   73.078   72.401   73.755
## 84     RGD14 37027 interpolated     2 0.986   31.228   30.863   31.592
## 85     RGc14 37027 interpolated     0 0.996  274.400  266.768  282.033
## 86     RGc14 37027 interpolated     1 0.996   24.518   24.418   24.619
## 87     RGc14 37027 interpolated     2 0.996   18.355   18.270   18.441
## 88     RGD15 37027 interpolated     0 0.987 1793.670 1777.615 1809.724
## 89     RGD15 37027 interpolated     1 0.987  203.796  201.493  206.100
## 90     RGD15 37027 interpolated     2 0.987   70.240   69.353   71.127
## 91     RGc15 37027 interpolated     0 0.997  234.673  225.851  243.495
## 92     RGc15 37027 interpolated     1 0.997   25.655   25.508   25.802
## 93     RGc15 37027 interpolated     2 0.997   18.394   18.269   18.519
## 94     RGD16 37027 interpolated     0 0.983 1539.874 1520.207 1559.540
## 95     RGD16 37027 interpolated     1 0.983   39.704   39.088   40.320
## 96     RGD16 37027 interpolated     2 0.983    9.644    9.523    9.765
## 97     RGc16 37027 interpolated     0 0.998  122.606  116.878  128.335
## 98     RGc16 37027 interpolated     1 0.998    2.358    2.345    2.371
## 99     RGc16 37027 interpolated     2 0.998    1.747    1.740    1.755
## 100    RGD17 37027 interpolated     0 0.993 1190.721 1176.273 1205.170
## 101    RGD17 37027 interpolated     1 0.993  126.164  124.455  127.873
## 102    RGD17 37027 interpolated     2 0.993   44.699   44.030   45.368
## 103    RGc17 37027 interpolated     0 0.997  380.131  373.375  386.886
## 104    RGc17 37027 interpolated     1 0.997   12.276   12.171   12.381
## 105    RGc17 37027 interpolated     2 0.997    6.641    6.604    6.679
## 106    RGD18 37027 interpolated     0 0.986 2304.240 2290.738 2317.742
## 107    RGD18 37027 interpolated     1 0.986  296.102  292.933  299.270
## 108    RGD18 37027 interpolated     2 0.986   76.031   75.068   76.993
## 109    RGc18 37027 interpolated     0 0.996  220.000  212.572  227.429
## 110    RGc18 37027 interpolated     1 0.996    4.727    4.704    4.750
## 111    RGc18 37027 interpolated     2 0.996    3.665    3.654    3.676
```

```
divestim.df <- divestim %>%
 mutate(habitat = str_to_title(design[as.character(site),"type"]))
```

Here is the resulting curve, showing the higher diversity in soil samples relative to the lake samples.

```
# divestim.df %>%
#   ggplot(aes(x = x, y = y,
#              ymin = y.lwr, ymax = y.upr,
#              color = habitat, fill = habitat, group = site)) +
#   geom_ribbon(data=subset(divestim.df, method == "extrapolated"), alpha = 0.3) +
#   geom_line(data=subset(divestim.df, method == "interpolated"), size = 1, alpha = .8) +
#   geom_line(alpha = 1, linetype = "dashed") +
#   scale_x_continuous(labels = scales::comma, limits = c(0, 90000)) +
```

```
#   labs(x = "Sample size", y = "Estimated richness") +
#   theme(legend.position = "none") +
#   #theme(legend.position =  c(.88,.5)) +
#   annotate(label = "Soil", size = 6, geom = "text", x = 85000, y = 5000) +
#   annotate(label = "Water", size = 6, geom = "text", x = 85000, y = 1500) +
#   scale_color_grey(end = .7) +
#   scale_fill_grey(end = .7)
```

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

```
# hill.estim <- divestim$AsyEst %>% filter(Diversity == "Species richness") %>%
#   left_join(rownames_to_column(alpha.div), by = c("Observed" = "S.obs")) %>%
#   select(Site, rowname, station, molecule, type, distance) %>%
#   left_join(divestim$AsyEst, by = "Site")

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

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

```
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)

# summary(hill.water.mod.rich)
# summary(hill.water.mod.shan)
# summary(hill.water.mod.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")
))
```

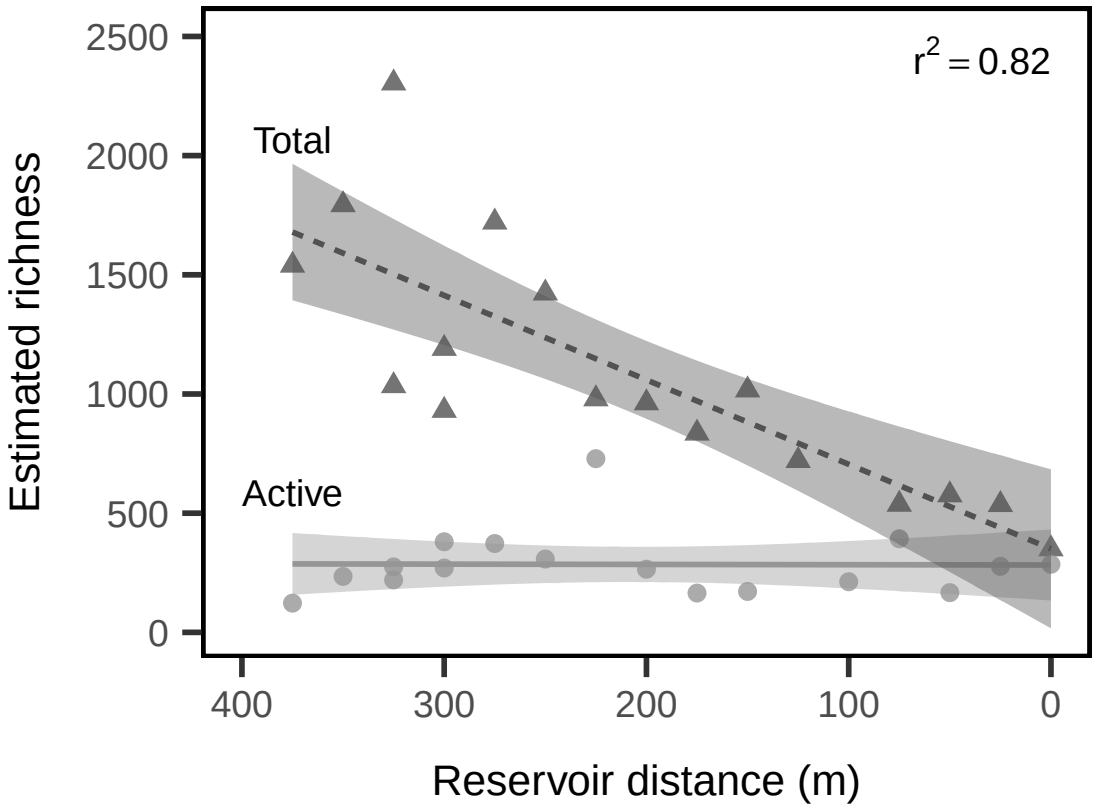
```
# Summary table of the model results.
hill.water.mods %>%
  group_by(Diversity) %>%
  rename("Term" = term,
         "Estimate" = estimate,
         "Std. Error" = std.error,
         "Statistic" = statistic,
         "p-value" = p.value) %>%
  select(Diversity, everything()) %>%
  pander(round =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

Diversity	Term	Estimate	Std. Error	Statistic	p-value
Richness	(Intercept)	350.5	121.5	2.885	0.0072

Diversity	Term	Estimate	Std. Error	Statistic	p-value
Richness	distance	3.544	0.5111	6.933	0
Richness	moleculeRNA	-67.95	170.7	-0.398	0.6935
Richness	distance:moleculeRNA	-3.531	0.7192	-4.91	0
Shannon	(Intercept)	55.27	24.9	2.22	0.0342
Shannon	distance	0.277	0.1048	2.644	0.0129
Shannon	moleculeRNA	-38.34	34.99	-1.096	0.2819
Shannon	distance:moleculeRNA	-0.2675	0.1474	-1.815	0.0796
Simpson	(Intercept)	31.56	7.659	4.12	3e-04
Simpson	distance	0.0522	0.0322	1.621	0.1154
Simpson	moleculeRNA	-23.75	10.76	-2.207	0.0351
Simpson	distance:moleculeRNA	-0.0382	0.0453	-0.8427	0.406

```
# hill.estim %>% filter(type == "water") %>%
#   mutate(molecule = ifelse(molecule == "DNA", "Total", "Active")) %>%
#   ggplot(aes(x = distance, y = Estimator,
#              ymin = LCL, ymax = UCL,
#              color = molecule, fill = molecule, shape = molecule)) +
#   geom_point(size =3) +
#   # geom_errorbar(size = .5, aes(ymin = Estimator - s.e., ymax = Estimator + s.e.),
#   #                width = 10, alpha = 0.5) +
#   geom_smooth(method = "lm", aes(linetype = molecule)) +
#   labs(x = "Reservoir distance (m)",
#        y = "") +
#   scale_color_manual(values = my.cols) +
#   scale_fill_manual(values = my.cols) +
#   theme(legend.position = c(.88,.95), strip.placement = "outside",
#         strip.text = element_text(size = 16)) +
#   scale_x_reverse() +
#   facet_grid(Diversity ~ ., scales = "free", switch = "y") +
#   guides(fill = guide_legend(override.aes=list(fill=NA)))
  #facet_grid(Diversity ~ ., scales = "free")

# postitions for labels
xpos = max((na.omit(hill.water$distance)))
yposDNA = predict(hill.water.mod.rich, newdata = data.frame(distance = 400, molecule = "DNA"))
yposRNA = predict(hill.water.mod.rich, newdata = data.frame(distance = 400, molecule = "RNA"))
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,
             color = molecule, fill = molecule, shape = molecule)) +
  geom_point(size =3, alpha = 0.8) +
  # geom_errorbar(size = .5, width = 10, alpha = 0.5) +
  geom_smooth(method = "lm", aes(linetype = molecule)) +
  labs(x = "Reservoir distance (m)",
       y = "Estimated richness") +
  scale_x_reverse(limits = c(400,0)) +
  scale_y_continuous(breaks = seq(0, 2500, by = 500)) +
  scale_color_manual(values = my.cols) +
  scale_fill_manual(values = my.cols) +
  theme(legend.position = "none") +
```

```
  guides(fill = guide_legend(override.aes=list(fill=NA))) +
  annotate("text", x = 375, y = yposRNA + 300,
           label = "Active", size = 5) +
  annotate("text", x = 375, y = yposDNA + 300,
           label = "Total", size = 5) +
  annotate(geom = "text", x = 0, y = 2500, hjust = 1, vjust = 1, size = 5,
           label = paste0("r^2== ",round(summary(hill.water.mod.rich)$r.squared, 2)), parse = T) +
  ggsave("figures/alpha_fig.pdf")
alpha.fig
```



So, from the basis of these results, we can make the following conclusions. First, we note that diversity in the total community decays from the stream inlet to the dam of the reservoir. That is, all the lines have a negative slope. However, we do not see this decay in the metabolically active community. Second, we note that the metabolically actively community has much lower diversity than the total community near the soils, but this difference decreases toward the dam. Last, because we quantified diversity across three orders of Hill numbers (q = 0, 1, and 2), we can also say something about the relative importance of rare versus common taxa along the reservoir transect. We see the the significance of the distance-by-molecule interaction term decrease as rare taxa are downweighted in favor of common taxa. This suggests that the differences between the active and total communities along the transect is driven primarily by rare taxa. However, the general trend of higher Simpson diversity across the whole transect suggests that low-activity, but relatively common, taxa are maintained in the reservoir.

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 = 400, molecule = c("RNA", "DNA")))
```

```
##          1          2
## 0.03070577 0.18754706
```

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

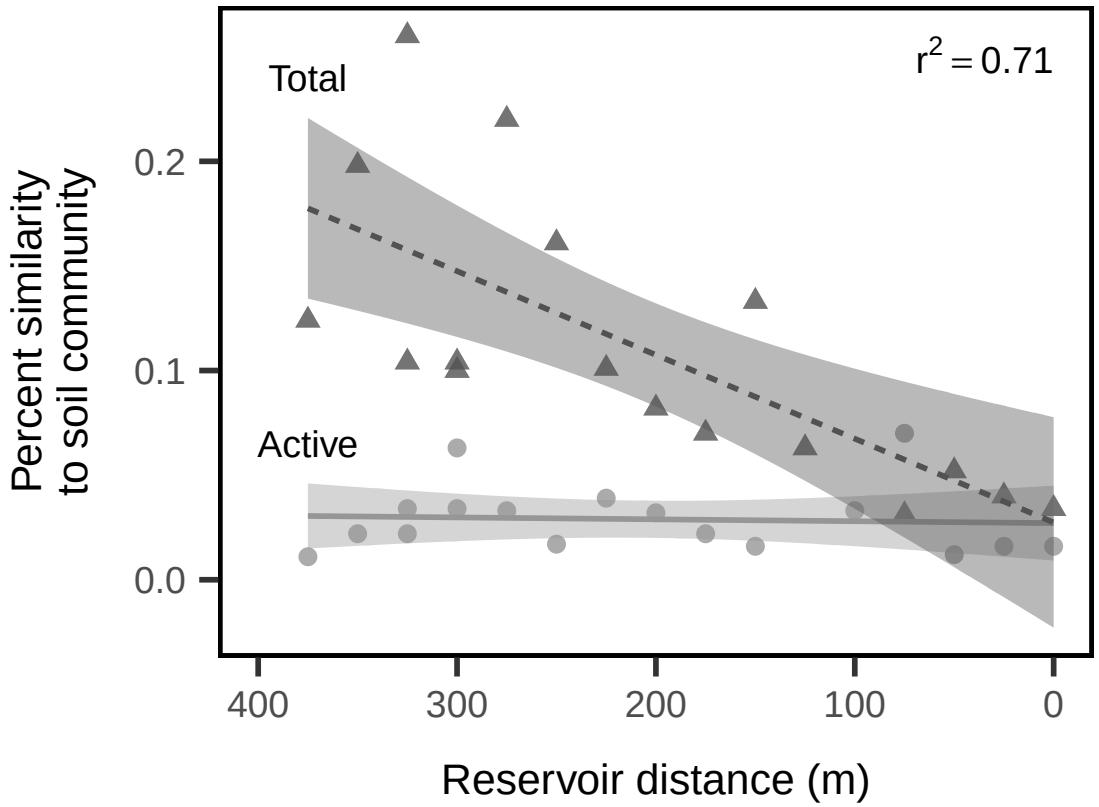
Table 2: Fitting linear model: bray.mean ~ distance * molecule

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.02739	0.01774	1.544	0.1331
distance	0.0004004	7.464e-05	5.365	8.319e-06
moleculeRNA	-0.0003186	0.02493	-0.01278	0.9899
distance:moleculeRNA	-0.0003913	0.000105	-3.726	0.000806

```
# # 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)
# D2.R2 <- round(summary(fit.Fig.3b)$r.squared, 2)
# summary(fit.Fig.3b)
#
#
# DNA.int.3b <- fit.Fig.3b$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.plot <- UL.sim %>%
  mutate(molecule = ifelse(UL.sim$molecule == "DNA", "Total", "Active")) %>%
  ggplot(aes(x = distance, y = bray.mean,
             color = molecule, fill = molecule, shape = molecule)) +
  geom_point(alpha = 0.8, size = 3, show.legend = T) +
  geom_smooth(method = "lm", show.legend = T, aes(linetype = molecule)) +
  labs(y = str_wrap("Percent similarity to soil community", width = 20),
       x = "Reservoir distance (m)") +
  scale_color_manual(values = my.cols) +
  scale_fill_manual(values = my.cols) +
  theme(legend.position = "none") +
  scale_x_reverse(limits = c(400,0)) +
  annotate(geom = "text", x = 0, y = max(UL.sim$bray.mean), hjust = 1, vjust = 1, size = 5,
```

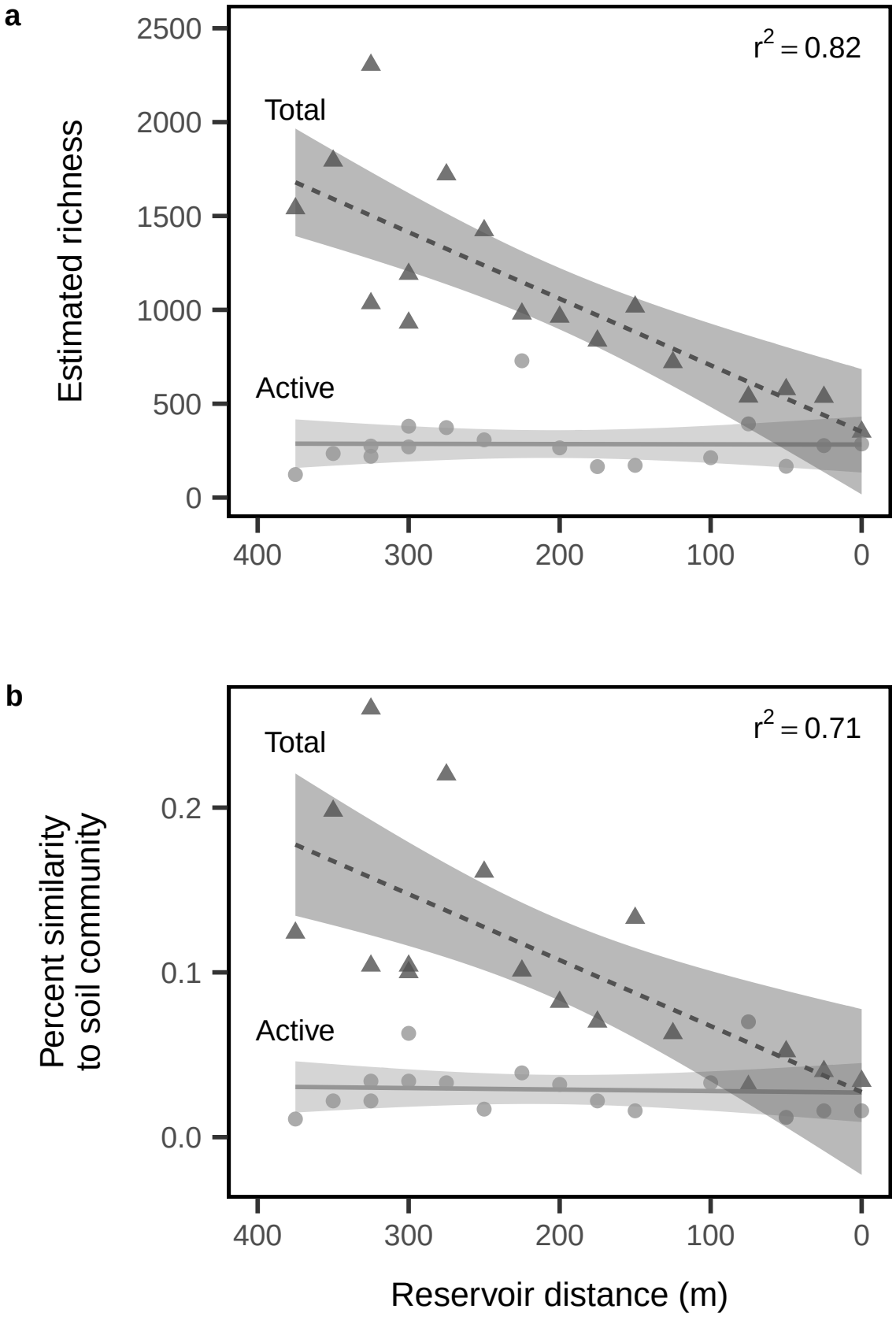
```
          label = paste0("r^2== ",round(summary(model.terr)$r.squared, 2)), parse = T) +
  annotate("text", x = 375, y = .065, label = "Active", size = 5) +
  annotate("text", x = 375, y = .24, label = "Total", size = 5) +
  ggsave("figures/similarity_fig.pdf")

similarity.plot
```



We find that our model captures most of the variation in community structure ($R^2 = 0.7084136$). We note a significant influence of distance on community similarity and the presence of a significant interaction between distance and whether the comparison is for active or total bacterial communities. This indicates that total communities decay faster with distance to soils than active communities do, which might be explained by the large difference in initial intercept. Active communities are always highly dissimilar to soil communities and remain so across the lake, while total lake communities are initially similar to soils, but this influence dissipates with distance into the reservoir.

Create combined figure

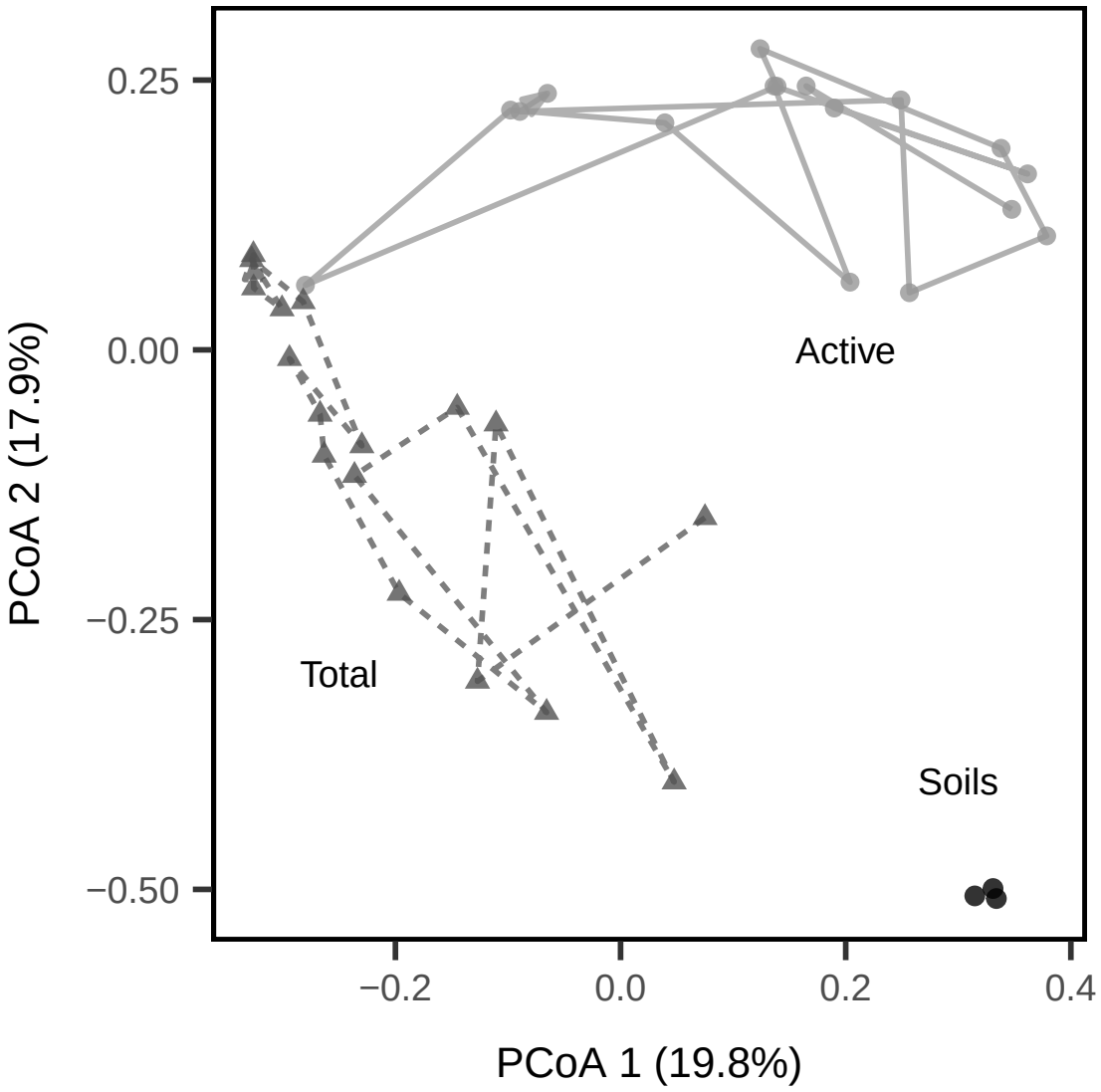
```
plot_grid(alpha.fig + labs(x = ""), similarity.plot,
          align = "hv",
          labels = "auto", ncol = 1) +
  ggsave("figures/alpha_similarity_paneled.pdf")
```


a
Estimated richness
2500
2000
1500
1000
500
0
$r^2 = 0.82$
Total
Active
400
300
200
100
0
b
Percent similarity
to soil community
0.2
0.1
0.0
$r^2 = 0.71$
Total
Active
400
300
200
100
0
Reservoir distance (m)

How does community structure change along the gradient?

First, we'll just get an overview of how the communities look along the aquatic transect.

```
ul.pcoa <- cmdscale(vegdist(OTUsREL.log, method="bray"), 2, eig = T, add = T)
explainvars <- round(eigenvals(ul.pcoa)[c(1,2)]/sum(eigenvals(ul.pcoa)),3) *100
water.pcvals <- data.frame(scores(ul.pcoa)) %>%
  rownames_to_column("name") %>%
  left_join(rownames_to_column(design, "name")) %>%
  arrange(desc(distance)) %>% filter(type == "water")
soil.pcvals <- data.frame(scores(ul.pcoa)) %>%
  rownames_to_column("name") %>%
  left_join(rownames_to_column(design, "name")) %>%
  arrange(desc(distance)) %>% filter(type == "soil")
pc_dists <- tibble(
  DNA_dim1 = subset(water.pcvals, molecule == "DNA")$Dim1,
  DNA_dim2 = subset(water.pcvals, molecule == "DNA")$Dim2,
  RNA_dim1 = subset(water.pcvals, molecule == "RNA")$Dim1,
  RNA_dim2 = subset(water.pcvals, molecule == "RNA")$Dim2)
pcoa.fig <- data.frame(scores(ul.pcoa)) %>%
  rownames_to_column("name") %>%
  left_join(rownames_to_column(design, "name")) %>%
  arrange(desc(distance)) %>% filter(type == "water") %>%
  mutate(molecule = ifelse(molecule == "DNA", "Total", "Active")) %>%
  ggplot(aes(x = Dim1, y = Dim2)) +
  geom_path(size = 1, alpha = 0.75, arrow = arrow(angle = 20,
                          length = unit(0.35, "cm"),
                          type = "closed"), aes(color = molecule, linetype = molecule)) +
  geom_point(size = 3, alpha = 0.8, aes(color = molecule, shape = molecule)) +
  geom_point(data = select(soil.pcvals, Dim1, Dim2), col = "black", alpha = .8, size = 3) +
  scale_color_manual("Community Subset", values = my.cols) +
  geom_segment(data = pc_dists,
               aes(x = DNA_dim1, y = DNA_dim2,
                   xend = RNA_dim1, yend = RNA_dim2),
               alpha = 0) +
  #coord_fixed(ratio = 1, xlim = c(-.4, .4)) +
  labs(x = paste0("PCoA 1 (", explainvars[1],"%)"),
       y = paste0("PCoA 2 (", explainvars[2],"%)")) +
  theme(legend.position = "none") +
  annotate(geom = "text", x = .2, y = 0, label = "Active", size = 5) +
  annotate(geom = "text", x = -.25, y = -.3, label = "Total", size = 5) +
  annotate(geom = "text", x = .3, y = -.4, label = "Soils", size = 5) +
  ggsave("figures/pcoa.pdf")
pcoa.fig
```



So, it appears that there is convergence in community structure along the path from stream inlet to the dam. This could reflect a loss of soil-derived taxa in the aquatic samples. To test this, we'll look at β-diversity along the gradient with respect to the soil samples. If we see a decay in similarity to soils, this suggests soil taxa are having a comparatively lower influence with distance from the inlet.

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)

# how many otus are active relative to the total number of otus
length(nvr.act) / ncol(lake.and.soil.total)
```

```
## [1] 0.8814706
```

```
length(yes.act) / ncol(lake.and.soil.total)
```

```
## [1] 0.1185294
```

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

```
## [1] 35765
```

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

```
## [1] 594544
```

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

```
## [1] 0.05674201
```

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

```
## [1] 0
```

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

```
## [1] 624979
```

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

```
## [1] 0
```

```
sum(rowSums(w.rna[,names(yes.act)])) / sum(rowSums(w.rna))
```

```
## [1] 0.9915438
```

```
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.

```
# 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$distance)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.21392 -0.10372 -0.02366  0.09693  0.26253
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          2.1199393  0.0774515  27.371 3.21e-14 ***
## design.dna$distance 0.0025505  0.0003258   7.828 1.12e-06 ***
## ---
## 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:  0.8034, Adjusted R-squared:  0.7902
## F-statistic: 61.28 on 1 and 15 DF,  p-value: 1.124e-06
```

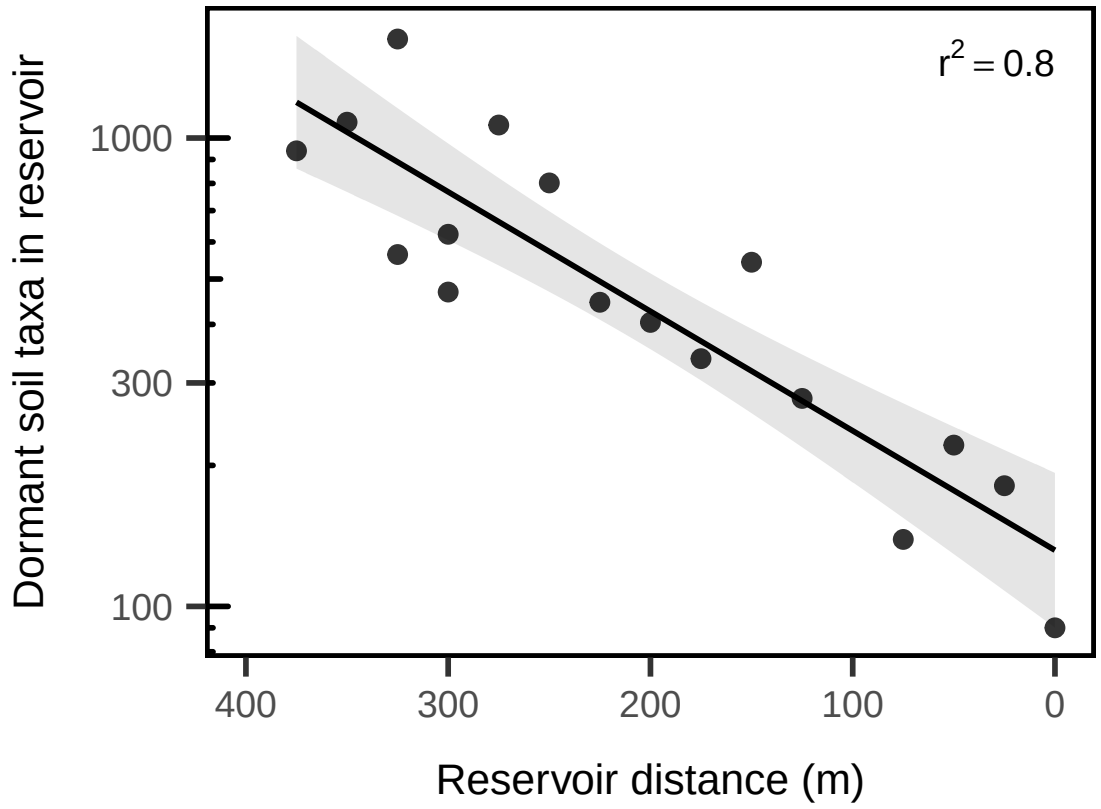
```
T1.R2 <- round(summary(terr.mod1)$r.squared, 2)
T1.int <- terr.mod1$coefficients[1]
T1.slp <- terr.mod1$coefficients[2]
pander(terr.mod1)
```

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)	2.12	0.07745	27.37	3.215e-14
design.dna$distance	0.002551	0.0003258	7.828	1.124e-06

```
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_x_reverse(limits = c(400,0)) +
  scale_y_log10() +
  annotation_logticks(sides = "l", size = 1) +
  labs(x = "Reservoir distance (m)",
```

```
      y = "Dormant soil taxa in reservoir") +
  annotate("text", x = 0, y = max(terr.rich), hjust = 1, vjust = 1, size = 5,
           label = paste0("r^2== ",T1.R2), parse = T)  +
  ggsave("figures/transients.pdf")
transient.plot
```




```
# plot_grid(alpha.fig,
#           similarity.plot,
#           pcoa.fig + ,
#           transient.plot,
#           align = "hv", axis = "tlbr",
#           labels = "auto", ncol = 2) +
#   ggsave("figures/large_panel.pdf", width = 12, height = 8)
```

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
```

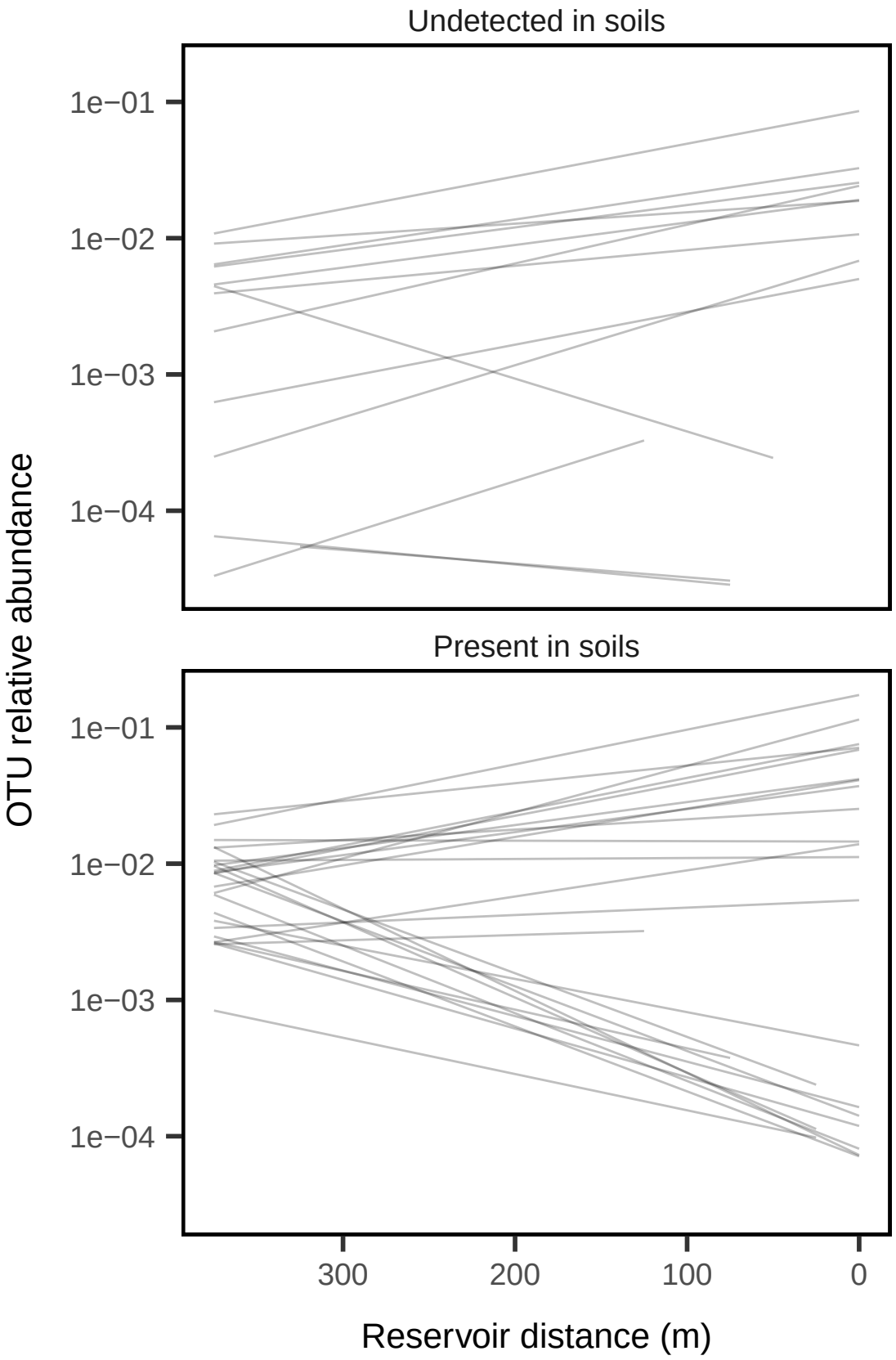
```
# define the 'core' taxa as otus present in 50% of samples
in.lake.core <- w.dna[, which((colSums(in.lake.rna.pa) / nrow(in.lake.rna.pa)) >= 0.75)]

# 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")
```

Now, lets plot the abundances of the OTUs across the reservoir and split them up into whether they were recovered in soils or not.

```
bind_rows(in.soil.to.plot, in.lake.to.plot) %>%
  ggplot(aes(x = distance, y = rel_abundance, group = otu_id)) +
  labs(x = "Reservoir distance (m)",
       y = "OTU relative abundance") +
  geom_line(alpha = 0.25, stat = "smooth", method = "lm", se = F, show.legend = F) +
  scale_y_log10() +
  scale_x_reverse() +
  facet_wrap(~ found, ncol = 1,
             labeller = as_labeller(c(
               `lake` = "Undetected in soils",
               `soils` = "Present in soils")))
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
```

```
## Warning: Removed 59 rows containing non-finite values (stat_smooth).
```



From this figure, we note a few important points. First, we observe that core reservoir taxa that are not detected in the soil samples tend to increase in relative abundance along the reservoir transect. We also note

that for the taxa that are present in the soil samples, some tend to increase drastically, while others tend to increase, along the transect. This suggests that there may be two classes of soil-derived OTUs that contribute to reservoir bacterial diversity:
- taxa where the reservoir is a sink (i.e., maintained via mass effects from the soils) - aquatic taxa seeded by populations stored in the soils

```
# 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
```

```
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
```

```
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
```

Now we will visualize the significant taxa

```
pander(nonsoil.core.decreasing, caption = "Core taxa not found in soils that get rarer along the transe
```

Table 4: Core taxa not found in soils that get rarer along the transect. (continued below)

OTU	slope	pval	Domain	Phylum
Otu00020	4.933e-07	0.9784	Bacteria	Proteobacteria
Otu00138	3.152e-05	0.04589	Bacteria	Firmicutes
Otu01010	1.364e-08	0.8511	Bacteria	Actinobacteria

OTU	slope	pval	Domain	Phylum
Otu01055	6.268e-08	0.25	Bacteria	Actinobacteria

Table 5: Table continues below

Class	Order	Family
Betaproteobacteria	Burkholderiales	Alcaligenaceae
Bacilli	Bacillales	Bacillaceae_1
Actinobacteria	Actinomycetales	Dermabacteraceae
Actinobacteria	Actinomycetales	Dietziaceae

Genus
Alcaligenaceae_unclassified
Bacillus
Brachybacterium
Dietzia

```
pander(nonsoil.core.increasing, caption = "Core taxa not found in soils that get more common along the
```

Table 7: Core taxa not found in soils that get more common along the transect. (continued below)

OTU	slope	pval	Domain	Phylum
Otu00004	-0.0001379	3.031e-06	Bacteria	Actinobacteria
Otu00007	-3.822e-06	0.704	Bacteria	Proteobacteria
Otu00023	-3.269e-07	0.7409	Bacteria	Proteobacteria
Otu00025	-5.193e-05	0.000563	Bacteria	Actinobacteria
Otu00032	-1.799e-05	0.2422	Bacteria	Bacteroidetes
Otu00038	-4.082e-05	0.0004677	Bacteria	Actinobacteria
Otu00040	-3.681e-05	1.522e-05	Bacteria	Proteobacteria
Otu00118	-7.165e-06	0.01503	Bacteria	Actinobacteria
Otu00156	-9.057e-06	0.0002607	Bacteria	Bacteria_unclassified

Table 8: Table continues below

Class	Order
Actinobacteria	Actinomycetales
Betaproteobacteria	Burkholderiales
Gammaproteobacteria	Pseudomonadales
Actinobacteria	Actinomycetales
Bacteroidetes_unclassified	Bacteroidetes_unclassified
Actinobacteria	Actinomycetales
Alphaproteobacteria	Rhodospirillales
Actinobacteria	Actinobacteria_unclassified
Bacteria_unclassified	Bacteria_unclassified

Family	Genus
Actinomycetales_unclassified	Actinomycetales_unclassified
Burkholderiaceae	Polynucleobacter
Moraxellaceae	Acinetobacter
Microbacteriaceae	Microbacteriaceae_unclassified
Bacteroidetes_unclassified	Bacteroidetes_unclassified
Actinomycetales_unclassified	Actinomycetales_unclassified
Acetobacteraceae	Roseomonas
Actinobacteria_unclassified	Actinobacteria_unclassified
Bacteria_unclassified	Bacteria_unclassified

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

Table 10: Core taxa found in soils that get rarer along the transect.
(continued below)

OTU	slope	pval	Domain	Phylum
Otu00009	4.862e-05	0.06326	Bacteria	Proteobacteria
Otu00012	7.397e-07	0.9069	Bacteria	Proteobacteria
Otu00018	4.823e-05	0.02295	Bacteria	Proteobacteria
Otu00022	9.926e-06	0.5565	Bacteria	Verrucomicrobia
Otu00028	2.89e-05	0.06155	Bacteria	Proteobacteria
Otu00030	1.849e-05	0.07112	Bacteria	Actinobacteria
Otu00039	6.677e-06	0.3738	Bacteria	Proteobacteria
Otu00042	1.432e-05	0.06336	Bacteria	Proteobacteria
Otu00059	5.62e-05	0.05497	Bacteria	Actinobacteria
Otu00065	4.952e-05	0.05262	Bacteria	Bacteroidetes
Otu00077	5.202e-05	0.0459	Bacteria	Bacteroidetes
Otu00081	2.039e-05	0.04586	Bacteria	Proteobacteria
Otu00086	1.075e-05	0.06061	Bacteria	Proteobacteria
Otu00095	3.442e-05	0.05901	Bacteria	Proteobacteria
Otu00545	2.144e-06	0.05592	Bacteria	Actinobacteria

Table 11: Table continues below

Class	Order	Family
Gammaproteobacteria	Pseudomonadales	Pseudomonadaceae
Betaproteobacteria	Burkholderiales	Comamonadaceae
Gammaproteobacteria	Pseudomonadales	Pseudomonadaceae
Opitutae	Opitutae_unclassified	Opitutae_unclassified
Gammaproteobacteria	Pseudomonadales	Pseudomonadaceae
Actinobacteria	Actinomycetales	Micrococcaceae
Betaproteobacteria	Burkholderiales	Comamonadaceae
Betaproteobacteria	Burkholderiales	Burkholderiaceae
Actinobacteria	Actinomycetales	Micrococcaceae
Sphingobacteriia	Sphingobacteriales	Sphingobacteriaceae
Flavobacteriia	Flavobacteriales	Flavobacteriaceae
Betaproteobacteria	Burkholderiales	Oxalobacteraceae
Alphaproteobacteria	Rhizobiales	Bradyrhizobiaceae
Betaproteobacteria	Burkholderiales	Comamonadaceae

Class	Order	Family
Actinobacteria	Solirubrobacterales	Solirubrobacteraceae

Genus
Pseudomonas
Comamonadaceae_unclassified
Pseudomonas
Opitutae_unclassified
Pseudomonas
Micrococcus
Comamonas
Burkholderia
Arthrobacter
Pedobacter
Flavobacterium
Janthinobacterium
Bradyrhizobium
Comamonadaceae_unclassified
Solirubrobacter

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

Table 13: Core taxa found in soils that get more common along the transect. (continued below)

OTU	slope	pval	Domain	Phylum
Otu00001	-2.297e-05	0.02728	Bacteria	Proteobacteria
Otu00002	-0.000238	0.0005166	Bacteria	Actinobacteria
Otu00003	-0.0001095	0.0003038	Bacteria	Verrucomicrobia
Otu00005	-5.261e-05	0.002303	Bacteria	Bacteroidetes
Otu00008	-4.242e-05	0.004938	Bacteria	Actinobacteria
Otu00010	-4.197e-05	0.5399	Bacteria	Proteobacteria
Otu00011	-2.029e-05	0.5457	Bacteria	Proteobacteria
Otu00014	-0.000103	0.000156	Bacteria	Actinobacteria
Otu00015	-0.0001461	5.141e-05	Bacteria	Actinobacteria
Otu00033	-1.544e-05	0.437	Bacteria	Proteobacteria

Table 14: Table continues below

Class	Order
Betaproteobacteria	Burkholderiales
Actinobacteria	Actinomycetales
Spartobacteria	Spartobacteria_unclassified
Sphingobacteriia	Sphingobacteriales
Actinobacteria	Actinomycetales
Proteobacteria_unclassified	Proteobacteria_unclassified
Betaproteobacteria	Betaproteobacteria_unclassified
Actinobacteria	Actinomycetales
Actinobacteria	Actinobacteria_unclassified

Class	Order
Alphaproteobacteria	Rhizobiales

Family	Genus
Comamonadaceae	Comamonadaceae_unclassified
Actinomycetales_unclassified	Actinomycetales_unclassified
Spartobacteria_unclassified	Spartobacteria_unclassified
Chitinophagaceae	Sediminibacterium
Actinomycetales_unclassified	Actinomycetales_unclassified
Proteobacteria_unclassified	Proteobacteria_unclassified
Betaproteobacteria_unclassified	Betaproteobacteria_unclassified
Actinomycetales_unclassified	Actinomycetales_unclassified
Actinobacteria_unclassified	Actinobacteria_unclassified
Rhizobiales_unclassified	Rhizobiales_unclassified

```
# p1 <- as.data.frame(OTUsREL[,nonsoil.core.increasing$OTU]) %>%
#   rownames_to_column("sampleID") %>%
#   left_join(rownames_to_column(design, "sampleID")) %>%
#   gather(OTU, rel_abund, -station, -molecule, -type, -distance, -sampleID) %>%
#   filter(molecule == "DNA") %>% left_join(OTU.tax) %>%
#   mutate(taxon = paste(Phylum, Class, Order, Family, Genus)) %>%
#   ggplot(aes(x = distance, y = rel_abund, group = OTU)) +
#   #geom_point(alpha = 0.5) +
#   geom_line(stat = "smooth", alpha = 0.5, size = 1,
#             color = "black", method = "loess", span = 1, se = FALSE) +
#   scale_x_reverse() +
#   scale_y_log10(labels = scales::scientific) +
#   theme(legend.position = "none") +
#   guides(color = guide_legend(ncol = 1)) +
#   labs(x = "",
#        y = "Relative Abundance",
#        title = "Absent from soil and significantly increasing")
#
# p2 <- as.data.frame(OTUsREL[,soil.core.increasing$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(taxon = paste(Class, Order)) %>%
#   ggplot(aes(x = distance, y = rel_abund, group = OTU)) +
#   #geom_point(alpha = 0.5) +
#   geom_line(stat = "smooth", alpha = 0.5, size = 1,
#             color = "black", method = "loess", span = 1, se = FALSE) +
#   scale_x_reverse() +
#   scale_y_log10(labels = scales::scientific) +
#   theme(legend.position = "none") +
#   guides(color = guide_legend(ncol = 1)) +
#   labs(x = "",
#        y = "Relative Abundance",
#        title = "Present in soil and significantly increasing")
#
```

```
# p3 <- 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(taxon = paste(Class, Order)) %>%
#   ggplot(aes(x = distance, y = rel_abund, group = OTU)) +
#   #geom_point(alpha = 0.5) +
#   geom_line(stat = "smooth", alpha = 0.5, size = 1,
#             color = "black", method = "loess", span = 1, se = FALSE) +
#   scale_x_reverse() +
#   scale_y_log10(labels = scales::scientific) +
#   theme(legend.position = "none") +
#   guides(color = guide_legend(ncol = 1)) +
#   labs(x = "Reservoir Transect (m)",
#        y = "Relative Abundance",
#        title = "Present in soil and significantly decreasing")
#
# cowplot::plot_grid(p1, p2, p3, align = "hv", labels = "AUTO", ncol = 1)

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
```

```
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
```

```
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
```

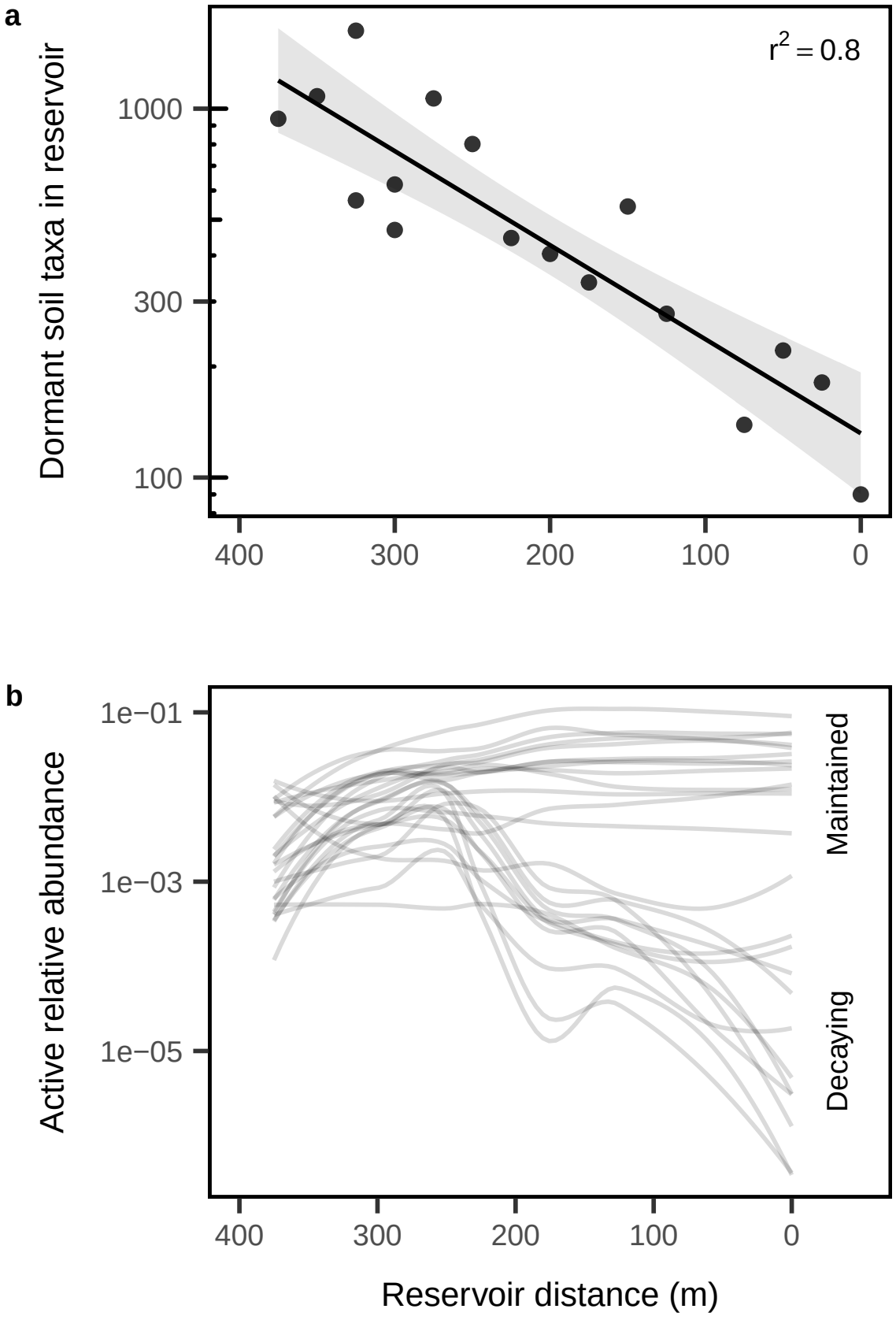
```
n3 <- length(unique(df3$OTU))
```

```
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
```

```
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 = 0.15, size = 1,
            method = "loess", span = .7, se = FALSE) +
  scale_x_reverse(limits = c(400, -50)) +
  scale_y_log10(labels = scales::scientific) +
  #theme(legend.position = "none") +
  #guides(color = guide_legend(ncol = 1)) +
  labs(x = "Reservoir distance (m)",
       y = "Active relative abundance") +
  annotate("text", x = -25, y = 1e-1, size = 5, hjust = 1, vjust = 1, angle = 90,
           label = "Maintained") +
  annotate("text", x = -25, y = 1e-5, size = 5, hjust = 0.5, vjust = 1, angle = 90,
           label = "Decaying") +
  ggsave("figures/taxa_origins.pdf")


# 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)
```

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


a
Dormant soil taxa in reservoir
1000
300
100
400
300
200
100
0
$r^2 = 0.8$
b
Active relative abundance
1e−01
1e−03
1e−05
Maintained
Decaying
400
300
200
100
0
Reservoir distance (m)

```
# soil.mods <- t(soil.core.mods) %>% as.data.frame()
# soil.mods$habitat <- "Present in soils"
# soil.mods <- soil.mods %>% rownames_to_column(var = "OTU")
# nonsoil.mods <- t(nonsoil.core.mods) %>% as.data.frame()
# nonsoil.mods$habitat <- "Absent from soils"
# nonsoil.mods <- nonsoil.mods %>% rownames_to_column(var = "OTU")
# rbind.data.frame(soil.mods, nonsoil.mods) %>%
#   filter(pval < 0.05) %>%
#   ggplot(aes(x = -slope, fill = habitat, color = habitat)) +
#   geom_line(stat = "density", alpha = 0.5, adjust = .8) +
#   geom_density(color = NA, adjust = .8, alpha = 0.2)
```

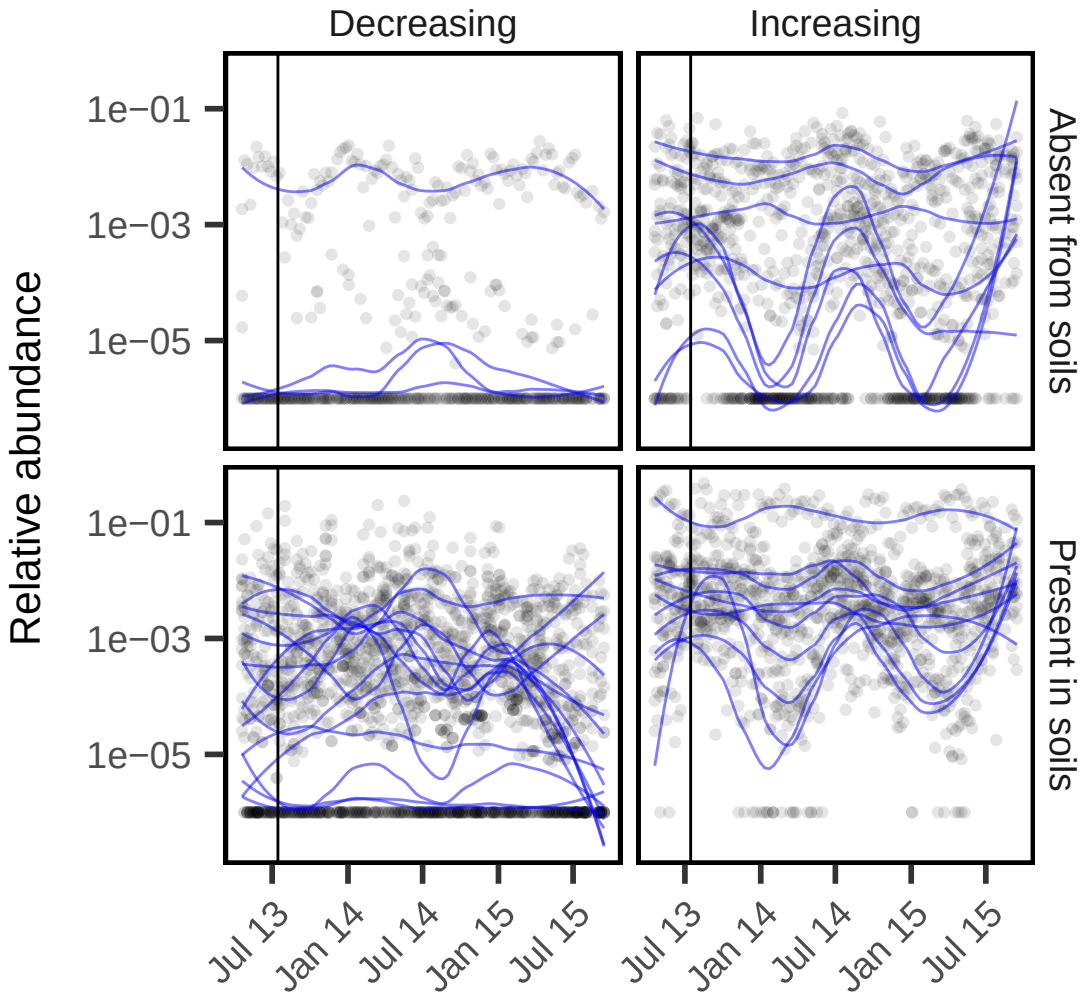
Are the "persistent" reservoir taxa really representative? Look over time...

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

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

# Subset to just the time series sites
UL.ts.OTUs <- total.OTUs[str_which(rownames(total.OTUs), "UL"),]

# make sure OTU table matches up with design order
UL.ts.design <- read_csv("data/UL_timeseries_design.csv")
UL.ts.OTUs <- UL.ts.OTUs[match(UL.ts.design$sample.name, rownames(UL.ts.OTUs)),]
UL.ts.OTUs.RNA <- decostand(UL.ts.OTUs[which(UL.ts.design$sample.type == "RNA"),], method = "total")
UL.ts.OTUs.DNA <- decostand(UL.ts.OTUs[which(UL.ts.design$sample.type == "DNA"),], method = "total")


env.ts.data <- read.table("data/ul-seedbank.env.txt", sep="\t", header=TRUE)
env.ts.data$date <- as.Date(parse_date_time(env.ts.data$date, "m d y"))
env.ts.data$doc[which(env.ts.data$doc == "**")] <- NA
env.ts.data$doc <- as.numeric(env.ts.data$doc)
summary(env.ts.data)
```

```
##    sample.id           date                temp            spc
##  Min.   :  1.00   Min.   :2013-04-19   Min.   : 2.21   Min.   :0.3300
##  1st Qu.: 31.75   1st Qu.:2013-11-20   1st Qu.: 5.50   1st Qu.:0.4600
##  Median : 62.50   Median :2014-06-23   Median :17.73   Median :0.5320
##  Mean   : 62.50   Mean   :2014-06-24   Mean   :16.18   Mean   :0.5172
##  3rd Qu.: 93.25   3rd Qu.:2015-01-25   3rd Qu.:25.05   3rd Qu.:0.5660
##  Max.   :124.00   Max.   :2015-09-14   Max.   :29.77   Max.   :0.6700
##                                        NA's   :2       NA's   :2
##      oxygen          salinity          secchi            ph
##  Min.   : 1.870   Min.   :0.1500   Min.   :0.200   Min.   : 6.890
##  1st Qu.: 5.237   1st Qu.:0.2200   1st Qu.:1.200   1st Qu.: 7.920
##  Median : 8.355   Median :0.2550   Median :1.600   Median : 8.415
##  Mean   : 8.961   Mean   :0.2487   Mean   :1.668   Mean   : 8.567
##  3rd Qu.:10.178   3rd Qu.:0.2700   3rd Qu.:2.200   3rd Qu.: 9.123
##  Max.   :22.240   Max.   :0.3200   Max.   :3.600   Max.   :10.860
```

```
##  NA's   :2        NA's   :2        NA's   :1       NA's   :2
##       chla              tp               tn              doc
##  Min.   :  0.92   Min.   :   8.26   Min.   : 0.407   Min.   :  2.00
##  1st Qu.: 12.63   1st Qu.:  26.30   1st Qu.: 0.882   1st Qu.: 32.25
##  Median : 37.67   Median :  34.85   Median : 1.210   Median : 61.50
##  Mean   : 79.25   Mean   :  84.25   Mean   : 1.889   Mean   : 61.57
##  3rd Qu.:121.31   3rd Qu.:  47.95   3rd Qu.: 1.490   3rd Qu.: 90.75
##  Max.   :523.56   Max.   :3200.00   Max.   :42.600   Max.   :121.00
##  NA's   :2        NA's   :2         NA's   :3        NA's   :2
##       orp             air.temp
##  Min.   :-41.800   Min.   :-11.60
##  1st Qu.:  9.325   1st Qu.:  7.00
##  Median : 21.700   Median : 18.50
##  Mean   : 50.507   Mean   : 15.57
##  3rd Qu.:104.975   3rd Qu.: 24.00
##  Max.   :225.200   Max.   : 32.00
##  NA's   :68        NA's   :2
```

```
UL.ts.design <- left_join(UL.ts.design, env.ts.data[,c("sample.id", "date")])
env.ts.data <- env.ts.data[-which(!(env.ts.data$date %in% UL.ts.design$date)),]

OTUs.in.core <- UL.ts.OTUs.RNA[, which(colnames(UL.ts.OTUs) %in% df.plot$OTU)]
cbind.data.frame(UL.ts.design[which(UL.ts.design$sample.type == "RNA"),], OTUs.in.core) %>% as_tibble()
  gather(-sample.name, -sample.type, -sample.id, -date, key = OTU, value = rel_abund) %>%
  mutate(soils = ifelse(OTU %in% unique(c(df2$OTU, df3$OTU)),
                        "Present in soils", "Absent from soils")) %>%
  mutate(change = ifelse(OTU %in% unique(c(df3$OTU, df4$OTU)),
                         "Decreasing", "Increasing")) %>%
  mutate(rel_abund = ifelse(rel_abund == 0, 1e-6, rel_abund)) %>%
  ggplot(aes(x = date, y = rel_abund, group = OTU)) +
  geom_point(alpha = .1) +
  geom_line(stat = "smooth", method = "loess", color = "blue",
            alpha = 0.5, span = .5, se = F) +
  geom_vline(aes(xintercept = as_date("2013-07-15"))) +
  scale_y_log10() +
  scale_x_date(labels = scales::date_format(format = "%b %y")) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  facet_grid(soils ~ change) +
  labs(x = "",
       y = "Relative abundance")
```



Many of them do appear to track the seasons quite well, suggesting there could be a seasonality component to the role of terrestrial inputs into the reservoir.

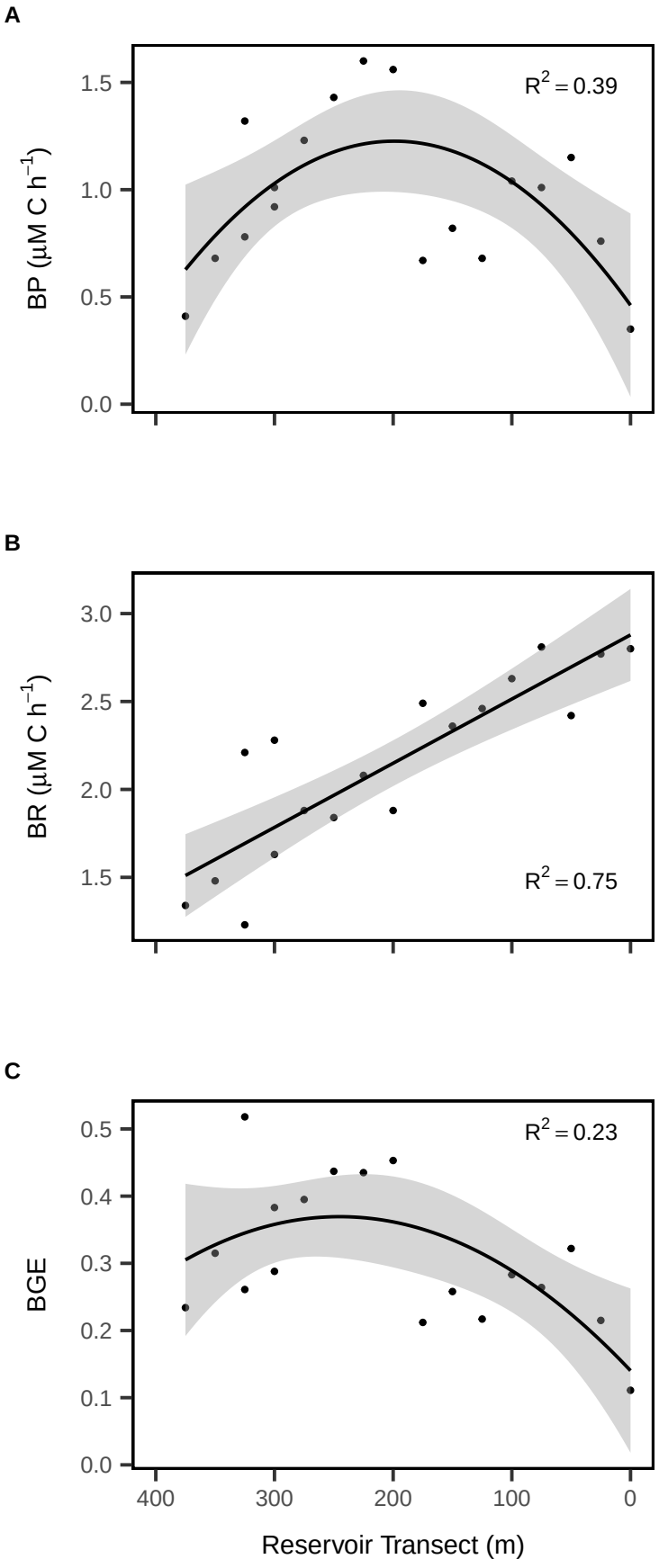
Not-included

Ecosystem functions

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


# Quadratic regression for BP
dist <- metab$dist
dist2 <- metab$dist^2
BP.fit <- lm(metab$BP ~ dist + dist2)
BP.R2 <- round(summary(BP.fit)$r.squared, 2)

# Simple linear regression for BR
```

```
BR.fit <- lm(metab$BR ~ metab$dist)
BR.R2 <- round(summary(BR.fit)$r.squared, 2)
BR.int <- BR.fit$coefficients[1]
BR.slp <- BR.fit$coefficients[2]

# Simple linear regression for BGE
BGE.fit <- lm(metab$BGE ~ metab$dist)
BGE.R2 <- round(summary(BGE.fit)$r.squared, 2)
BGE.int <- BGE.fit$coefficients[1]
BGE.slp <- BGE.fit$coefficients[2]

BP.R2
BR.R2
BGE.R2

BP.plot <- ggplot(metab, aes(x = dist, y = BP)) +
  geom_point() +
  geom_smooth(method = "lm", formula = y ~ x + I(x^2), color = "black") +
  annotate(geom = "text", x = 50, y = 1.5, size = 5,
           label = paste0("R^2== ",BP.R2), parse = T) +
  labs(y = expression(paste('BP (', mu ,'M C h'^-1* ')')),
       x = "Reservoir Transect (m)") +
  scale_x_reverse(limits = c(400,0))
BR.plot <- ggplot(metab, aes(x = dist, y = BR)) +
  geom_point() +
  geom_smooth(method = "lm", formula = y ~ x, color = "black") +
  annotate("text", x = 50, y = 1.5, size = 5,
           label = paste0("R^2== ",BR.R2), parse = T ) +
  labs(y = expression(paste('BR (', mu ,'M C h'^-1* ')')),
       x = "Reservoir Transect (m)") +
  scale_x_reverse(limits = c(400,0))
BGE.plot <- ggplot(metab, aes(x = dist, y = BGE)) +
  geom_point() +
  geom_smooth(method = "lm", formula = y ~ x + I(x^2), color = "black") +
  annotate("text", x = 50, y = .5, size = 5,
           label = paste0("R^2== ",BGE.R2), parse = T ) +
  labs(y = "BGE",
       x = "Reservoir Transect (m)") +
  scale_x_reverse(limits = c(400,0))
```

```
plot_grid(BP.plot + theme(axis.title.x = element_blank(), axis.text.x = element_blank(),
                          plot.margin = unit(c(1, 1, -1, 0), "cm")),
          BR.plot + theme(axis.title.x = element_blank(), axis.text.x = element_blank(),
                          plot.margin = unit(c(-1, 1, -1, 0), "cm")),
          BGE.plot + theme(plot.margin = unit(c(-1, 1, 0, 0), "cm")),
          align = "hv", ncol = 1, labels = "AUTO")
```


A
1.5
1.0
0.5
0.0
BP (µM C h^{-1})
$R^2 = 0.39$
B
3.0
2.5
2.0
1.5
BR (µM C h^{-1})
$R^2 = 0.75$
C
0.5
0.4
0.3
0.2
0.1
0.0
BGE
$R^2 = 0.23$
400
300
200
100
0
Reservoir Transect (m)

Relation of ecosystem functions and community structure

```
# detrend the spatial signal
bp.resid <- resid(lm(BP ~ dist + I(dist)^2, data = metab))
br.resid <- resid(lm(BR ~ dist, data = metab))


metab.resids <- metab
metab.resids$BR_resid <- br.resid + mean(metab$BR)
metab.resids$BP_resid <- bp.resid + mean(metab$BP)

transient.metabolism <- data.frame(transients = terr.REL, dist = design.dna$distance) %>%
  left_join(metab.resids)


bp.mod.quad <- lm(BP_resid ~ transients + I(transients^2), data = transient.metabolism)
bp.mod.lin <- lm(BP_resid ~ transients, data = transient.metabolism)
bp.mod.int <- lm(BP_resid ~ 1, data = transient.metabolism)
anova(bp.mod.int, bp.mod.lin, bp.mod.quad)
AIC(bp.mod.quad, bp.mod.lin, bp.mod.int)

br.mod.quad <- lm(BR_resid ~ transients + I(transients^2), data = transient.metabolism)
br.mod.lin <- lm(BR_resid ~ transients, data = transient.metabolism)
br.mod.int <- lm(BR_resid ~ 1, data = transient.metabolism)
anova(br.mod.int, br.mod.lin, br.mod.quad)
AIC(br.mod.int, br.mod.lin, br.mod.quad)

bge.mod.quad <- lm(BGE ~ transients + I(transients^2), data = transient.metabolism)
bge.mod.lin <- lm(BGE ~ transients, data = transient.metabolism)
bge.mod.int <- lm(BGE ~ 1, data = transient.metabolism)
anova(bge.mod.int, bge.mod.lin, bge.mod.quad)
AIC(bge.mod.int, bge.mod.lin, bge.mod.quad)

round(summary(br.mod.quad)$r.squared, 2)
round(summary(bp.mod.quad)$r.squared, 2)


total_core <- rowSums(OTUsREL[design$molecule == "DNA" & design$type == "water",
                              subset(rbind.data.frame(high.activity.water.core,
                                                      high.activity.soil.core), RNA.max > .01)$OTU])

summary(lm(BP ~ transients * dist, transient.metabolism))
summary(lm(BR ~ transients * dist, transient.metabolism))


data.frame(
  soil_core = rowSums(OTUsREL[design$molecule == "DNA" & design$type == "water",
           subset(soil.vs.lake.abunds, RNA.max > .01)$OTU]),
  dist = design.dna$distance) %>%
  left_join(metab.resids) %>% select(-BGE, -BP, -BR) %>% gather(metab, value, -soil_core, -dist) %>%
  ggplot(aes(x = soil_core, y = value, color = metab, fill = metab)) +
  geom_point(size = 2) +
  geom_smooth(alpha = .25, method = 'lm', formula = y ~ x + I(x^2)) +
```

```
  labs(x = "Relative Abundance of Soil-derived Core",
       y = expression(paste('Metabolism (', mu ,'M C h'^-1* ')'))) +
  scale_color_viridis("Ecosystem Function", discrete = T, begin = .1, end = .6, option = "D") +
  scale_fill_viridis("Ecosystem Function", discrete = T, begin = .1, end = .6, option = "D") +
  ggsave("figures/06_soilcore-function.pdf", bg = "white", width = 7, height = 6)

data.frame(
  water_core = rowSums(OTUsREL[design$molecule == "DNA" & design$type == "water",
                               subset(high.activity.water.core, RNA.max > .01)$OTU]),
  dist = design.dna$distance) %>%
  left_join(metab.resids) %>% select(-BGE,-BR,-BP) %>% gather(metab, value, -water_core, -dist) %>%
  ggplot(aes(x = water_core, y = value, color = metab, fill = metab)) +
  geom_point(size = 2) +
  geom_smooth(alpha = .25, method = 'lm', formula = y ~ x + I(x^2)) +
  labs(x = "Relative Abundance of non-soil-derived Core",
       y = expression(paste('Metabolism (', mu ,'M C h'^-1* ')'))) +
  scale_color_viridis("Ecosystem Function", discrete = T, begin = .1, end = .6, option = "D") +
  scale_fill_viridis("Ecosystem Function", discrete = T, begin = .1, end = .6, option = "D") +
  ggsave("figures/06_nonsoilcore-function.pdf", bg = "white", width = 7, height = 6)



data.frame(transients = resid(lm(terr.REL ~ design.dna$distance)) + mean(terr.REL), dist = design.dna$di
  left_join(metab.resids) %>% select(-BGE, -BP, -BR) %>% gather(metab, value, -transients, -dist) %>%
  ggplot(aes(x = transients, y = value, color = metab, fill = metab)) +
  geom_point(size = 2, show.legend = F) +
  geom_smooth(alpha = .25, method = 'lm', formula = y ~ x, show.legend = F) +
  annotation_logticks(sides = "b") +
  labs(x = "Relative Abundance of Transient Taxa",
       y = expression(paste('Metabolism (', mu ,'M C h'^-1* ')'))) +
  scale_color_viridis("Ecosystem Function", discrete = T, begin = .1, end = .6, option = "D") +
  scale_fill_viridis("Ecosystem Function", discrete = T, begin = .1, end = .6, option = "D") +
  scale_y_continuous(limits = c(0,3)) +
  theme(plot.margin = unit(c(1,1,0,0), "cm")) +
  ggsave("figures/06_transients-function.pdf", bg = "white", width = 7, height = 6)


core.metab <- data.frame(
  total_core = rowSums(OTUsREL[design$molecule == "DNA" & design$type == "water",
                               subset(rbind.data.frame(high.activity.water.core,
                                                       high.activity.soil.core), RNA.max > .01)$OTU]),
  dist = design.dna$distance) %>%
  left_join(metab.resids)

summary(lm(BP ~ total_core * dist, core.metab))
summary(lm(BR ~ total_core + dist, core.metab))


core.metab <- data.frame(
  total_core = rowSums(OTUsREL[design$molecule == "DNA" & design$type == "water",
                               subset(rbind.data.frame(high.activity.water.core,
                                                       high.activity.soil.core), RNA.max > .01)$OTU]),
  dist = design.dna$distance) %>%
```

```
  left_join(metab.resids)
core.metab$total_core_resid <- resid(lm(total_core ~ dist + I(dist^2), core.metab)) + mean(core.metab$t
summary(lm(BP_resid ~ total_core, core.metab))
summary(lm(BR_resid ~ total_core + I(total_core^2), core.metab))


core.metab %>% select(-BGE, -BP, -BR, -total_core) %>% gather(metab, value, -total_core_resid, -dist) %
  ggplot(aes(x = total_core_resid, y = value, color = metab, fill = metab)) +
  geom_point(size = 2, show.legend = F) +
  geom_smooth(alpha = .25, method = 'lm', formula = y ~ x, show.legend = F) +
  labs(x = "Relative Abundance of Core Taxa",
       y = expression(paste('Metabolism (', mu ,'M C h'^-1* ')'))) +
  scale_color_viridis("Ecosystem Function", discrete = T, begin = .1, end = .6, option = "D") +
  scale_fill_viridis("Ecosystem Function", discrete = T, begin = .1, end = .6, option = "D") +
  scale_y_continuous(limits = c(0,3)) +
  theme(plot.margin = unit(c(1,1,0,0), "cm")) +
  ggsave("figures/06_core-function.pdf", bg = "white", width = 7, height = 6)
```