

# Environmetal effects on diversity and composition

Luyao Kang

2023-07-02

## Contents

Data input . . . . .	1
Test the environmetanl effects on the microbial diversity . . . . .	1
Difference in environmental factors among layers . . . . .	1
Test the environmental effects on the community structure . . . . .	5

## Data input

Set work directory

```
setwd('e:/permafrost/')
wd_16s <- file.path(getwd(),"data/16S")
# if (!dir.exists(wd_16s)) {
#   dir.create(wd_16s)
# }
wd_fun <- file.path(getwd(),"data/metagenome")
save.dir <- file.path(getwd(),"result")
```

Read data

```
source("script/read_data.R")
```

## Test the environmetanl effects on the microbial diversity

Loading packages

```
library(ggplot2)
library(tidyr)
library(dplyr)
library(tibble)
library(ggpubr)
```

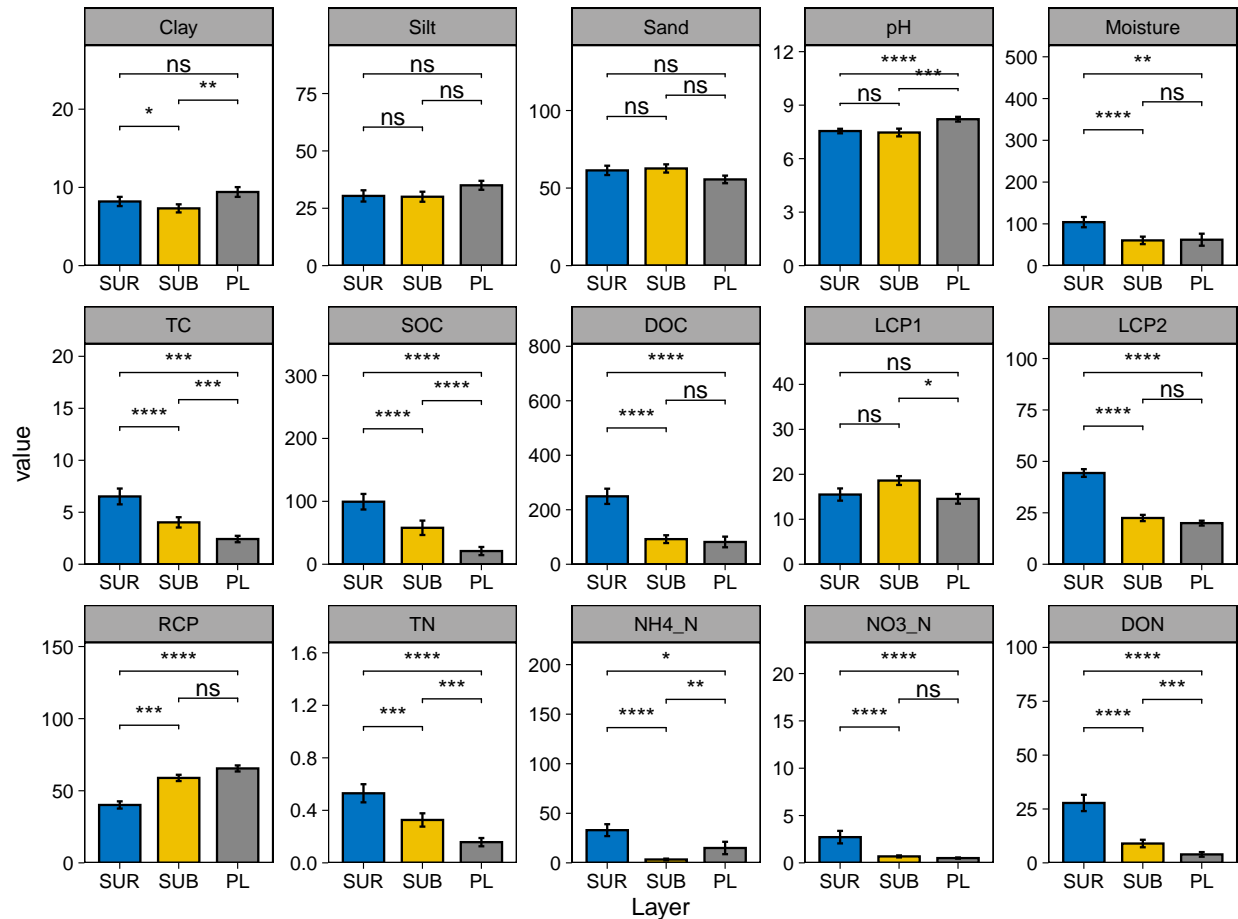
## Difference in environmental factors among layers

```

my_comparisons <- list( c("SUR", "SUB"), c("SUB", "PL"), c("SUR", "PL") )
env_vars <- c("Layer", "Clay", "Silt", "Sand", "pH", "Moisture", "TC", "SOC", "DOC", "LCP1", "LCP2", "R
meta_dat %>% as.tibble() %>% select(env_vars) %>%
  group_by(Layer) %>%
  summarise_all(list(mean = mean, se = plotrix::std.error)) %>%
  write.csv(., file = file.path(save.dir, "tables/env_comparison/env_compare.csv"))

p <- meta_dat %>% as.tibble() %>% select("Layer", env_vars) %>%
  pivot_longer(cols = -c(Layer), names_to = "env", values_to = "value") %>%
  mutate(Layer = factor(Layer, levels = c("SUR", "SUB", "PL"))) %>%
  mutate(env = factor(env, levels = env_vars)) %>%
  ggbarplot(x = "Layer", y = "value", fill = "Layer",
    facet.by = "env", palette = "jco",
    add = "mean_se", position = position_dodge(0.8)) +
  facet_wrap(vars(env), scales = "free", ncol = 5) +
  scale_y_continuous(expand = expansion(mult = c(0, 0.15))) +
  stat_compare_means(comparisons=my_comparisons, paired = TRUE, label = "p.signif",
    test = "wilcox.test", size = 4, p.adjust.method = "BH",
    step.increase = 0.22) +
  theme_linedraw() +
  theme(strip.text = element_text(color = 'black'),
    strip.background = element_rect(fill = "#aaa9a9"),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    legend.position = "none")
# save plot
# plot.name <- paste0(save.dir, "/figs/env_effect/env_paired_wilcox_test_abc.pdf")
# print(plot.name)
# cairo_pdf(filename = plot.name, width = 8.6, height = 7, onefile = TRUE)
# p
# dev.off()
p

```



Prepare the table for correlation analysis

```
alpha_div <- estimate_richness(phylo_rare, measures = c("Observed", "Chao1", "Shannon", "Simpson"))
library(picante)
pd <- pd(t(otu), tree, include.root = F) # estimate the phylogenetic diversity
alpha_div <- cbind(Layers = metadata$Layer, alpha_div, Faith = pd$PD,
  Evenness = alpha_div$Shannon/log(alpha_div$Observed)) %>%
  mutate(Layers = factor(Layers, levels = c('SUR', 'SUB', 'PL')))
cor_div_env_tab <- data.frame(metadata[-c(1, 2, 4:6)], Evenness = alpha_div$Evenness)
# You can use sel_env to specify the variables you want to use and sel_env_label to specify the labels for
sel_env <- c("MAT", "AI", "Moisture", "pH", "Clay", "Silt", "Sand", "TC",
  "SOC", "DOC", "TN", "NH4_N", "NO3_N", "DON",
  "LCP1", "LCP2", "RCP")
```

Correlation analysis

```
groups <- cor_div_env_tab$Layer
x <- data.frame(cor_div_env_tab[, sel_env])
y <- data.frame(Evenness = cor_div_env_tab$Evenness)
# You can use kendall, spearman, or pearson below:
method <- "pearson"
# Now calculate the correlation between individual Taxa and the environmental data
df <- NULL
```

```

for (i in colnames(x)) {
  for (j in colnames(y)) {
    for (k in unique(groups)) {
      a <- x[groups == k, i, drop = F]
      b <- y[groups == k, j, drop = F]
      tmp <- c(i, j, cor(a[complete.cases(b), ], b[complete.cases(b), ], use = "everything", method = m
        cor.test(a[complete.cases(b), ], b[complete.cases(b), ], method = method)$p.value, k)
      if (is.null(df)) {
        df <- tmp
      } else {
        df <- rbind(df, tmp)
      }
    }
  }
}

df <- data.frame(row.names = NULL, df)
colnames(df) <- c("Env", "div", "r", "Pvalue", "Layer")
df$Pvalue <- as.numeric(as.character(df$Pvalue))
df$AdjPvalue <- rep(0, dim(df)[1])
df$r <- as.numeric(as.character(df$r))

# Now we generate the labels for signifant values
df$Significance <- cut(df$Pvalue, breaks = c(-Inf, 0.001, 0.01, 0.05, Inf), label = c("****", "***", "**",
# We ignore NAs
df <- df[complete.cases(df), ]
df <- df[, c("Env", "Layer", "r", "Significance")]
# We want to reorganize the Env data based on they appear
df$Env <- factor(df$Env, levels = sel_env)
df$Layer <- factor(df$Layer, levels = rev(c("SUR", "SUB", "PL")))

```

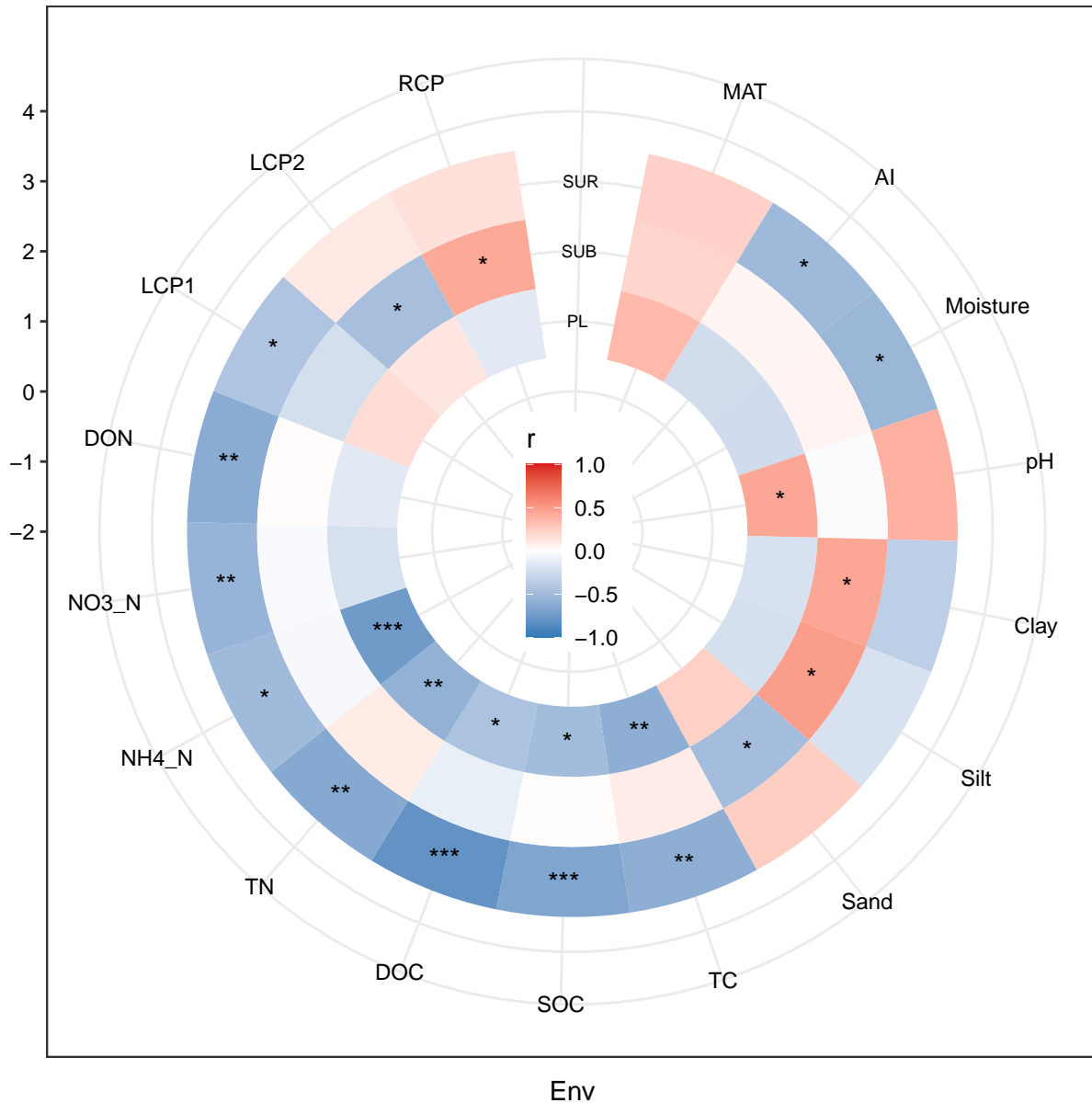
Heatmap for alpha diversity and environmental factors

```

row_num = length(levels(df$Layer))
p <- ggplot(aes(x = Env, y = as.numeric(Layer), fill = r), data = df) +
  xlim(c("", sel_env)) +
  ylim(c(-row_num/1.5, row_num + 1)) +
  geom_tile() + ylab("") +
  scale_fill_gradient2(low = "#2C7BB6", mid = "white", high = "#D7191C", limits = c(-1, 1)) +
  annotate(x = "", y = 1:row_num, label = levels(df$Layer), size = 2.5, geom = "text") +
  geom_text(aes(label = Significance), color = "black", size = 4)

p <- p + coord_polar(start = -0.15, theta = "x") +
  theme_bw() +
  theme(axis.text = element_text(color = "black"),
        legend.position = c(0.5, 0.5),
        legend.key.size = unit(0.5, "cm"))
# ggsave(file.path(save.dir, './figs/env_effect/env_alpha_plot.pdf'), p, width = 9, height = 6, units =
p

```



## Test the environmental effects on the community structure

Partial mantel test function for the relationship between the envs and community

```
partial.mantel.fun <- function(phylo) {
  env.table <- data.frame(sample_data(phylo))
  #env.table <- env.table[complete.cases(env.table), ]
  otu_table <- as.matrix(t(otu_table(phylo)))
  otu_table_hel <- decostand(otu_table, 'hellinger')
  otu_table_hel_dist <- vegdist(otu_table_hel, 'bray', upper=F)
  df <- NULL
  vars <- c("MAT", "AI", "Moisture", "pH", "Clay", "Silt", "Sand", "TC",
            "SOC", "DOC", "TN", "NH4_N", "NO3_N", "DON",
```

```

      "LCP1", "LCP2", "RCP")
for (x in vars) {
  y.dist <- vegdist(scale(env.table[,x]), 'euclidean', na.rm = T)
  z.dist <- vegdist(scale(env.table[, setdiff(vars, x)]), 'euclidean', na.rm = T)
  mode <- mantel.partial(otu_table_hel_dist, y.dist, z.dist,
                        method = "pearson", permutations = 999, na.rm = T)

  r <- mode$statistic
  p <- mode$signif
  tmp <- data.frame(env = x, r = r, p.value = p)
  if(is.null(df))
    df <- tmp
  else
    df <- rbind(df, tmp)
}
return(df)
}

phylo_sur <- subset_samples(phylo, Layer == 'SUR')
phylo_sur <- prune_taxa(taxa_sums(phylo_sur)>=1, phylo_sur)
phylo_sub <- subset_samples(phylo, Layer == 'SUB')
phylo_sub <- prune_taxa(taxa_sums(phylo_sub)>=1, phylo_sub)
phylo_pl <- subset_samples(phylo, Layer == 'PL')
phylo_pl <- prune_taxa(taxa_sums(phylo_pl)>=1, phylo_pl)

set.seed(123)
par.mant.sur <- partial.mantel.fun(phylo_sur)
set.seed(123)
par.mant.sub <- partial.mantel.fun(phylo_sub)
set.seed(123)
par.mant.pl <- partial.mantel.fun(phylo_pl)

```

PLOT using ggcor package

```

## devtools::install_github('hannet91/ggcor')
library(ggcor)
par.man.tibble <- tibble(spec = c(rep('SUR', nrow(par.mant.sur)), rep('SUB', nrow(par.mant.sub)), rep('PL', nrow(par.mant.pl))),
                        r = c(rbind(par.mant.sur, par.mant.sub, par.mant.pl)))

vars <- c("MAT", "AI", "Moisture", "pH", "Clay", "Silt", "Sand", "TC",
          "SOC", "DOC", "TN", "NH4_N", "NO3_N", "DON",
          "LCP1", "LCP2", "RCP")
env.table <- sample_data(phylo)[, vars]
mantel02 <- par.man.tibble %>%
  mutate(r = cut(r, breaks = c(-Inf, 0.2, 0.5, Inf),
                 labels = c("<0.20", "0.20-0.5", ">0.50"),
                 right = FALSE),
         p.value = cut(p.value, breaks = c(-Inf, 0.001, 0.01, 0.05, Inf),
                      labels = c("<0.001", "0.001-0.01", "0.01-0.05", ">0.05"),
                      right = T))
p <- quickcor(env.table, type = "lower") + geom_square() +
  add_link(mantel02, mapping = aes(colour = p.value, size = r),
          diag.label = TRUE) +
  scale_color_manual(values = c('#d95f02', '#1b9e77', '#3c5488ff', 'grey')) +

```

```

scale_size_manual(values = c(0.5, 2, 3)) +
geom_diag_label() + remove_axis("x")
# ggsave(file.path(save.dir, './figs/env_effect/partail_matel_plot.pdf'), p, width = 9, height = 6, uni
p

```

