Environmetal effects on diversity and composition

Luyao Kang

2023-07-02

Table of Contents

## 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")

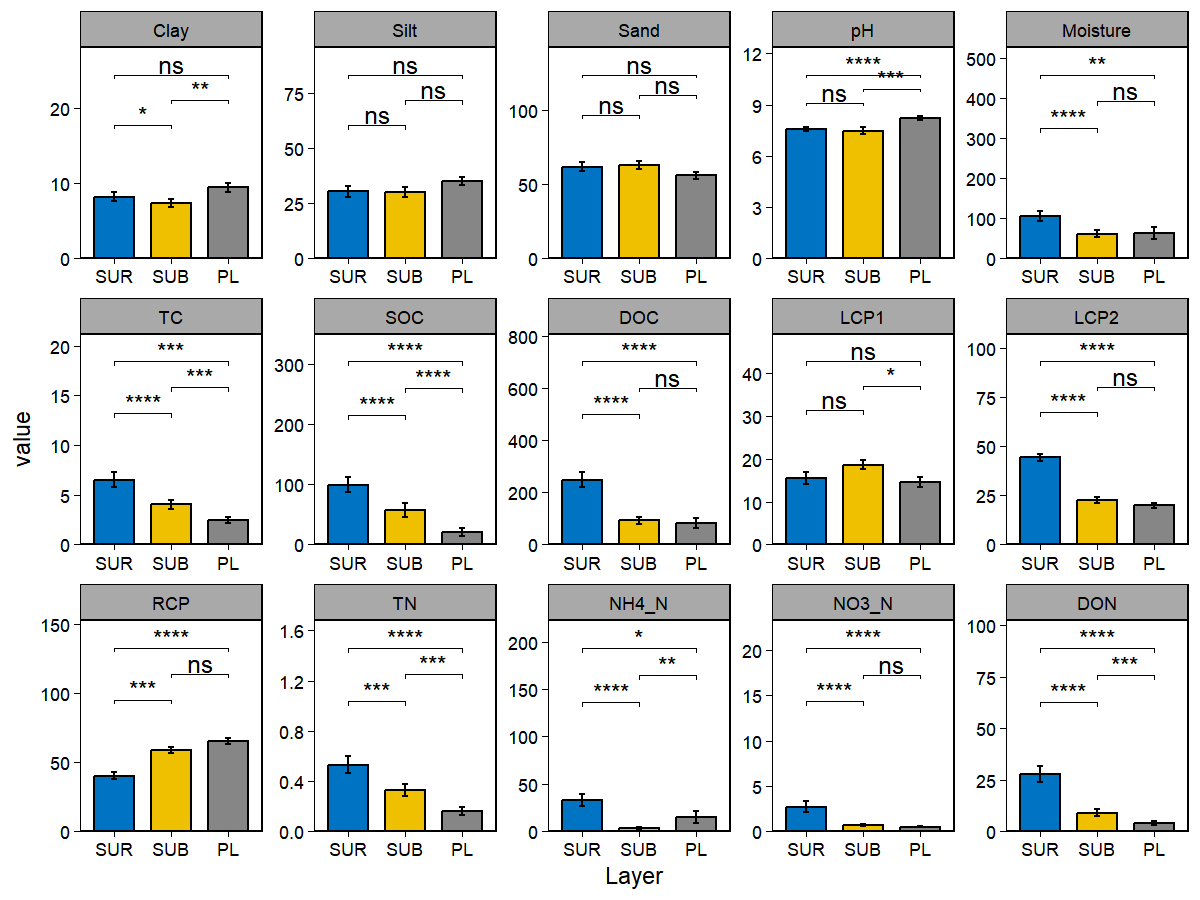
## Test the environmetanl effects on the microbial diversity

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", "RCP", "TN", "NH4\_N", "NO3\_N", "DON")  
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

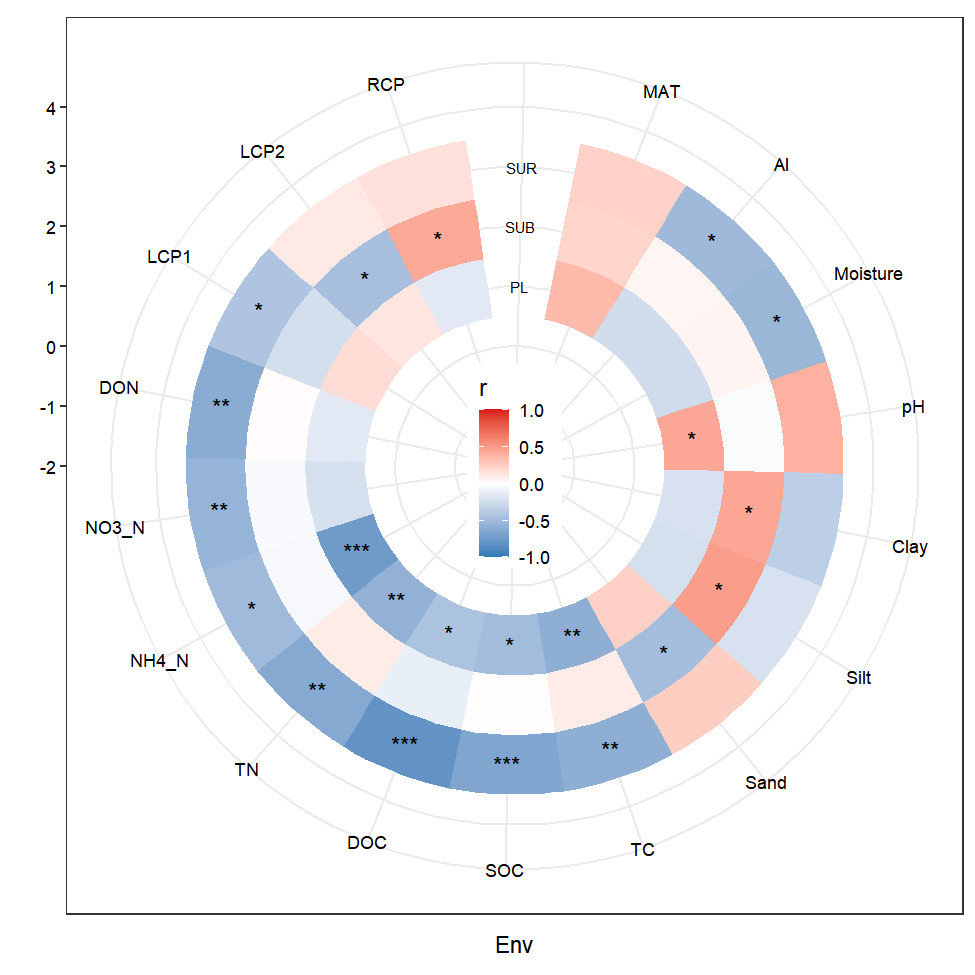
alpha\_div <- estimate\_richness(phylo\_rare, measures = c("Observed", "Chao1", 'Shannon', 'Simpson'))  
library(picante)  
pd <- pd(t(otu), tree, include.root = F) # estimate the phylogenetic diversity  
alpha\_div <- cbind(Layers = metadata$Layer, alpha\_div, Faith = pd$PD,  
 Evenness = alpha\_div$Shannon/log(alpha\_div$Observed)) %>%  
 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 labes for the pannel  
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 = method),   
 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 = "in")  
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))),   
 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, units = "in")  
p

