02 Community bulk properties

Angel Rain

01/03/2024

Contents

1 Setting up the workspace

1.1 Loading Packages

```
rm(list = ls())
library(cowplot)
library(egg)
library(readxl)
library(FactoMineR)
library(factoextra)
```

```
library(RColorBrewer)  #Expando color palette
library(dplyr)
library(stringr)  # FOr editing string
library(reshape2)  #For 'melt' function
library(rstatix)
library(nlme)  #Mixed linear models
library(ggsignif)  #Stars in plots
library(tinytex)
library(ggtext)
library(kableExtra)  #Pretty tables
library(olsrr)  # For normality
```

1.2 Loading colorblind palette

```
# Setting the colorblind palette
cbp1 <- c("#999999", "#FFDB6D", "#E69F00", "#56B4E9", "#009E73", "#F0E442",
    "#0072B2", "#D55E00", "#CC79A7", "#293352")
```

1.3 Load functional data

```
## Load functional data
Funct.data <- read.csv("../data/CommunityProperties_CycleExp.csv", header = T)
tibble(Funct.data)
```

```
## # A tibble: 160 x 8
##    Date     Sample.ID Cycle Microcosm Abundance_106cellmL Biomass_mgCL
##    <chr>    <chr>     <chr>     <int>               <dbl>        <dbl>
##  1 10.10.19 C01       C0            1                3.01       NA
##  2 10.10.19 C010      C0           10                2.67        0.993
##  3 10.10.19 C011      C0           11                1.34        1.4
##  4 10.10.19 C012      C0           12                6.12        0.125
##  5 10.10.19 C013      C0           13                6.89        0.769
##  6 10.10.19 C014      C0           14                5.12        0.325
##  7 10.10.19 C015      C0           15                2.54        0.323
##  8 10.10.19 C016      C0           16                3.49        1.50
##  9 10.10.19 C017      C0           17                5.76        1.59
## 10 10.10.19 C018      C0           18                3.15        0.385
## # i 150 more rows
## # i 2 more variables: ConsoCellobiose_mgCL <dbl>, CUE <dbl>
```

1.4 Add community types from Scritp01

```
# Add microcosms per community types (from Script01)
Acido = c("6", "9", "15", "19", "20")
Aero = c("1", "3", "7")
Pseu = c("2", "13", "14")
Othe = c("4", "5", "8", "10", "11", "12", "16", "17", "18")
```

```
# Create empty vector for clustering identification
Funct.data$cluster = NA
Funct.data$cluster[Funct.data$Microcosm %in% Acido] = "Acidovorax-type"
Funct.data$cluster[Funct.data$Microcosm %in% Aero] = "Aeromonas-type"
Funct.data$cluster[Funct.data$Microcosm %in% Pseu] = "Pseudomonas-type"
Funct.data$cluster[Funct.data$Microcosm %in% Othe] = "Others"

Funct.data <- Funct.data %>%
    mutate_at(vars(Cycle, Microcosm, cluster), list(factor))

# Re-order community type levels
Funct.data$cluster <- factor(Funct.data$cluster, c("Acidovorax-type", "Aeromonas-type",
    "Pseudomonas-type", "Others"))

# Create a numerical ID for incubation cycles
Funct.data$Time = as.numeric(str_remove(Funct.data$Cycle, "C"))
tibble(Funct.data)
```

```
## # A tibble: 160 x 10
##    Date     Sample.ID Cycle Microcosm Abundance_106cellmL Biomass_mgCL
##    <chr>    <chr>     <fct> <fct>                   <dbl>        <dbl>
##  1 10.10.19 C01       C0    1                        3.01       NA
##  2 10.10.19 C010      C0    10                       2.67        0.993
##  3 10.10.19 C011      C0    11                       1.34        1.4
##  4 10.10.19 C012      C0    12                       6.12        0.125
##  5 10.10.19 C013      C0    13                       6.89        0.769
##  6 10.10.19 C014      C0    14                       5.12        0.325
##  7 10.10.19 C015      C0    15                       2.54        0.323
##  8 10.10.19 C016      C0    16                       3.49        1.50
##  9 10.10.19 C017      C0    17                       5.76        1.59
## 10 10.10.19 C018      C0    18                       3.15        0.385
## # i 150 more rows
## # i 4 more variables: ConsoCellobiose_mgCL <dbl>, CUE <dbl>, cluster <fct>,
## #   Time <dbl>
```

```
# Transform CUE to percentage
Funct.data$CUE <- Funct.data$CUE * 100
```

2 Community properties by community types over cycles

2.1 Community properties bulk plots

```
# template plot to extract legend
plot.template <- ggplot(data = Funct.data[Funct.data$Cycle != "C7", ], aes(x = (Cycle),
    y = Abundance_106cellmL, fill = cluster)) + geom_boxplot(alpha = 0.5, outlier.shape = NA,
    size = 0.3) + theme_bw() + scale_fill_manual(values = c("grey", "blue",
    "orange", "white"), name = "") + theme(legend.position = "top", legend.text = element_text(size = 5
    margin = margin(0, 0, 0, 0.02, unit = "cm")), legend.margin = margin(0,
    -10, 0, 12), legend.spacing.x = unit(0, "cm"))
```

```
# Get legend
legend.panel <- get_plot_component(plot.template, "guide-box-top", return_all = TRUE)

# Create a common features for plots
theme.plots <- theme_bw() + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()
    axis.title.y = element_markdown(hjust = 0.5), text = element_text(size = 7),
    legend.position = "none")

# Cell abundance
ylabels <- c("Abundance<br>(x10<sup>6</sup> Cells mL<sup>-1</sup>)", "Biomass<br>(mgC L<sup>-1</sup>)",
    "Cellobiose consumption<br>(mgC L<sup>-1</sup>)", "CUE")

# Prepare figures
comm.prop.plots <- list()
for (i in 1:4) {
    df.tmp <- Funct.data[, c(3, 4 + i, 9)]  #Subset data by variable
    names(df.tmp)[2] <- "Variable"
    comm.prop.plots[[i]] <- ggplot(data = df.tmp[df.tmp$Cycle != "C7", ], aes(x = (Cycle),
        y = Variable, fill = cluster)) + geom_boxplot(alpha = 0.5, outlier.shape = NA,
        size = 0.25) + geom_point(position = position_jitterdodge(jitter.width = 0.2,
        dodge.width = 0.8), alpha = 0.8, shape = 21, size = 1, stroke = 0.25) +
        scale_colour_manual(values = c("grey", "blue", "orange", "white", "lightblue"),
            name = "") + scale_fill_manual(values = c("grey", "blue", "orange",
        "white", "lightblue"), name = "") + geom_boxplot(data = df.tmp[df.tmp$Cycle ==
        "C7", ], aes(x = (Cycle), y = Variable, fill = Cycle), alpha = 0.5,
        outlier.shape = NA, width = 0.3, size = 0.25) + geom_point(data = df.tmp[df.tmp$Cycle ==
        "C7", ], aes(x = (Cycle), y = Variable, fill = Cycle), position = position_jitterdodge(jitter.w
        dodge.width = 0.8), alpha = 0.8, shape = 21, size = 1, stroke = 0.25) +
        labs(y = ylabels[i], x = NULL) + theme.plots + theme(axis.text.x = element_blank(),
        plot.margin = margin(t = 0, b = 0, r = 0.5, l = 0.5, unit = "cm"))

}
# Edit bottom panel
comm.prop.plots[[4]] <- comm.prop.plots[[4]] + theme(axis.text.x = element_text(),
    plot.margin = margin(t = 0, b = 0, r = 0.5, unit = "cm")) + labs(x = "Cycle")

# remove objects
rm(plot.template, theme.plots, ylabels)
```

2.2 Statistical analysis

```
# Empty dataframe for results
df.rmANOVA <- data.frame(Comm.Prop = names(Funct.data)[c(5:8)], HOV = NA, Normality = NA,
    CommType.F = NA, CommType.p = NA, Cycle.F = NA, Cycle.p = NA, CommTypexCycle.F = NA,
    CommTypexCycle.p = NA)

for (i in 1:4) {
    # Subet dataset
    tmp.df <- Funct.data[, c("Cycle", "cluster", "Microcosm", df.rmANOVA$Comm.Prop[i])]
    tmp.df <- (subset(tmp.df, Cycle %in% c("C1", "C2", "C3", "C4", "C5", "C6")))
    # repeated measurements ANOVA
```

```
    tmp.aov <- (aov(tmp.df[, 4] ~ Cycle * cluster + Error(Microcosm), data = tmp.df))
    # Test assumptions
    df.rmANOVA$Normality[i] <- ols_test_normality(tmp.aov[["Within"]][["residuals"]])[[1]][2] >
        0.05
    df.rmANOVA$HOV[i] <- levene_test(tmp.df[, 4] ~ cluster * Cycle, data = tmp.df)[4] >
        0.05

    # Save stats
    df.rmANOVA$CommType.F[i] <- summary(tmp.aov)[["Error: Microcosm"]][[1]]["cluster",
        4]
    df.rmANOVA$CommType.p[i] <- summary(tmp.aov)[["Error: Microcosm"]][[1]]["cluster",
        5]
    df.rmANOVA$Cycle.F[i] <- summary(tmp.aov)[["Error: Within"]][[1]]["Cycle ",
        4]
    df.rmANOVA$Cycle.p[i] <- summary(tmp.aov)[["Error: Within"]][[1]]["Cycle ",
        5]
    df.rmANOVA$CommTypexCycle.F[i] <- summary(tmp.aov)[["Error: Within"]][[1]]["Cycle:cluster",
        4]
    df.rmANOVA$CommTypexCycle.p[i] <- summary(tmp.aov)[["Error: Within"]][[1]]["Cycle:cluster",
        5]
}
```

Table summary statistical analyses for the repeated measurements ANOVA applied to the bulk community properties.

Table 1: Repeated measurement ANOVA for bulk properties

Variable	Assumptions		rmANOVA					
.	HOV	Normality	Comm.Type		Cycle		Comm.Type x Cycle	
			Fstat	P	Fstat	P	Fstat	P
Abundance_106cellmL	TRUE	TRUE	1.274	0.317	4.887	0.001	2.055	0.021
Biomass_mgCL	TRUE	TRUE	0.976	0.429	3.046	0.014	0.528	0.918
ConsoCellobiose_mgCL	TRUE	TRUE	1.699	0.207	1.780	0.126	0.495	0.937
CUE	TRUE	TRUE	0.441	0.727	8.266	0.000	0.896	0.571

3 Mixed linear model for bulk community properties

3.1 Normalizing values

```
# Cell abudances
Funct.data$Abundance_106cellmL.norm <- Funct.data$Abundance_106cellmL/max(Funct.data$Abundance_106cellml
    c("C7", "C0")])

# Biomasses
Funct.data$biomass.norm <- Funct.data$Biomass_mgCL/max(Funct.data$Biomass_mgCL[!Funct.data$Cycle %in%
    c("C7", "C0")])

# Cellobiose consumption
Funct.data$ConsoCellobiose_mgCL.norm <- Funct.data$ConsoCellobiose_mgCL/max(Funct.data$ConsoCellobiose_m
    c("C7", "C0")])
```

```
# Carbon use efficiency (CUE)
Funct.data$CUE.norm <- Funct.data$CUE/max(Funct.data$CUE[!Funct.data$Cycle %in%
    c("C7", "C0")], na.rm = T)
```

3.2 Statistical analysis

```
list.lmm.cluster <- list()
var.names <- names(Funct.data)[11:14]

for (i in 1:4) {
    # Set dataset
    tmp.df <- Funct.data[, c("Time", "cluster", "Microcosm", var.names[i], "Cycle")]
    tmp.df <- (subset(tmp.df, Cycle %in% c("C1", "C2", "C3", "C4", "C5", "C6")))
    names(tmp.df)[4] <- "Variable"
    # Save results linear mixed models by community type
    tmp.lmm <- lme(Variable ~ cluster + Time:cluster, random = ~1 | Microcosm,
        data = tmp.df, na.action = na.omit)
    results.lmm <- data.frame(coef(summary(tmp.lmm)))[5:8, ]
    results.lmm$variable <- var.names[i]
    # Save results linear mixed models all data
    tmp.lmm = lme(Variable ~ Time, random = ~1 | Microcosm, data = tmp.df, na.action = na.omit)
    results.lmm["All", ] <- c(coef(summary(tmp.lmm))[2, ], var.names[i])
    list.lmm.cluster[[i]] <- results.lmm
}

df.lmm.coef <- do.call(rbind, list.lmm.cluster)
df.lmm.coef$Category = c("Acidovorax", "Aeromonas", "Pseudomonas", "Others",
    "All")

df.lmm.coef$Value = as.numeric(df.lmm.coef$Value)
df.lmm.coef$Std.Error = as.numeric(df.lmm.coef$Std.Error)
df.lmm.coef$p.value = as.numeric(df.lmm.coef$p.value)
df.lmm.coef$Category = as.factor(df.lmm.coef$Category)
df.lmm.coef$Category = factor(df.lmm.coef$Category, c("All", "Others", "Pseudomonas",
    "Aeromonas", "Acidovorax"))

# Sort out dataset
df.lmm.coef$variable = as.factor(df.lmm.coef$variable)
df.lmm.coef$variable = recode(df.lmm.coef$variable, Abundance_106cellmL.norm = "Cell",
    biomass.norm = "Biomass", ConsoCellobiose_mgCL.norm = "Cellobiose", CUE.norm = "CUE")

# Add significance values
df.lmm.coef$symbol = NA
df.lmm.coef$symbol[df.lmm.coef$p.value < 0.05] <- "*"
df.lmm.coef$symbol[df.lmm.coef$p.value < 0.01] <- "**"
df.lmm.coef$symbol[df.lmm.coef$p.value < 0.001] <- "***"
```

4 Effect sizes plots

```
# Categories for analysis
plots.effsize.list <- list()  #Empty list

labs2 <- levels(df.lmm.coef$Category)
labs2
```

```
## [1] "All"         "Others"      "Pseudomonas" "Aeromonas"   "Acidovorax"
```

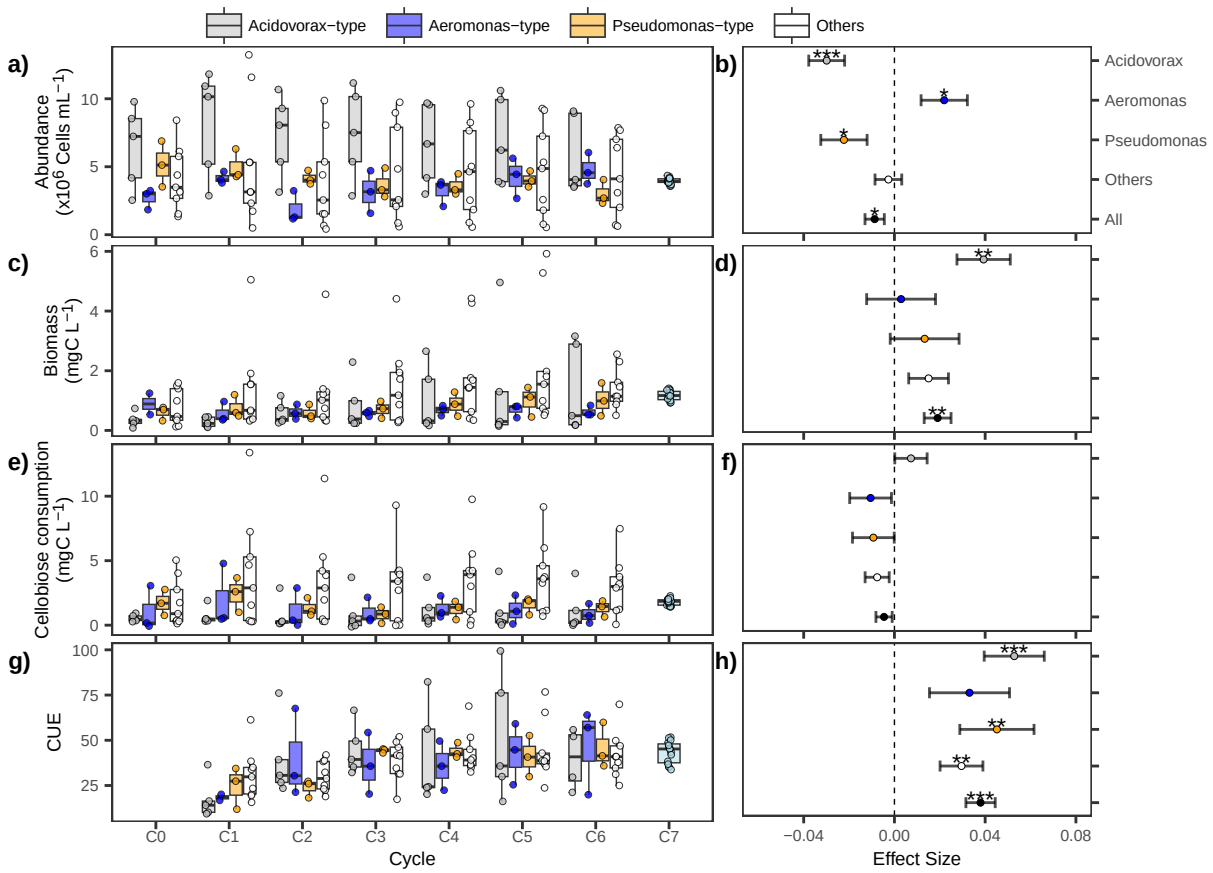
```
for (i in 1:length(levels(df.lmm.coef$variable))) {
    plots.effsize.list[[i]] <- df.lmm.coef[df.lmm.coef$variable == levels(df.lmm.coef$variable)[i],
        ] %>%
        ggplot(aes(Value, as.numeric(Category), fill = Category)) + geom_vline(xintercept = 0,
        linetype = "dashed", size = 0.25) + geom_errorbar(aes(y = as.numeric(Category),
        xmin = Value + Std.Error, xmax = Value - Std.Error), width = 0.3, size = 0.5,
        color = "black", alpha = 0.7) + geom_point(size = 1.1, color = "black",
        shape = 21, stroke = 0.2) + theme_bw() + theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank()) + xlab("") + ylab(NULL) + scale_fill_manual(values = c("bla
        "white", "orange", "blue", "grey"), name = "") + geom_text(aes(y = as.numeric(Category),
        x = Value, label = symbol), size = 3, hjust = 0.5, vjust = 0.3) + scale_y_continuous(breaks = NU
        labels = NULL, sec.axis = sec_axis(~., breaks = 1:length(labs2), labels = NULL)) +
        theme(plot.margin = margin(t = 0, b = -0.55, r = 0.5, unit = "cm")) +
        xlim(-0.06, 0.08) + theme(text = element_text(size = 7)) + theme(legend.position = "none")
}

# Add community type labels
plots.effsize.list[[1]] <- plots.effsize.list[[1]] + scale_y_continuous(breaks = NULL,
    labels = NULL, sec.axis = sec_axis(~., breaks = 1:length(labs2), labels = labs2))
# Edit x labels for bottom plot
plots.effsize.list[[4]] <- plots.effsize.list[[4]] + theme(plot.margin = margin(t = 0,
    b = 0, r = 0.5, unit = "cm")) + xlab("Effect Size")
```

4.1 Panel functional response Cycle all

Bulk community properties, separated by community type. Left panels. Abundances, biomass during 6 cycles of biological interaction (C1 to C6) and the homogenizing dispersal event (C7) in the three microcosms community types and the set of unique assemblages. Right panels display slopes derived from the mixed linear model for a, e) cell abundances, b, f) cellobiose consumption c, g), biomass and d, h) carbon use efficiency (CUE). Left panel dashed lines present the average functional value at C6. Right panel dashed lines indicate slope equal to 0.

```
plot_grid(plot_grid(legend.panel, NULL, rel_widths = c(0.8, 0.2)), plot_grid(plot_grid(comm.prop.plots[
    comm.prop.plots[[2]], comm.prop.plots[[3]], comm.prop.plots[[4]], ncol = 1,
    axis = "rl", align = "v", rel_heights = c(0.8, 0.8, 0.8, 0.91), labels = c("a)",
        "c)", "e)", "g)"), label_size = 9), NULL, plot_grid(plots.effsize.list[[1]],
    plots.effsize.list[[2]], plots.effsize.list[[3]], plots.effsize.list[[4]],
    ncol = 1, axis = "rl", align = "v", rel_heights = c(0.8, 0.8, 0.8, 0.91),
    labels = c("b)", "d)", "f)", "h)"), label_size = 9, label_x = -0.01, hjust = 1),
    ncol = 3, axis = "l", align = "v", rel_widths = c(0.9, -0.01, 0.6)), ncol = 1,
    rel_heights = c(0.05, 1))
```




```
pdf("../Figures/Figure4.pdf", width = 4.33071, height = 5)

plot_grid(plot_grid(legend.panel, NULL, rel_widths = c(0.8, 0.2)), plot_grid(plot_grid(comm.prop.plots[
    comm.prop.plots[[2]], comm.prop.plots[[3]], comm.prop.plots[[4]], ncol = 1,
    axis = "rl", align = "v", rel_heights = c(0.8, 0.8, 0.8, 0.91), labels = c("a)",
        "c)", "e)", "g)"), label_size = 9, label_x = 0.01, label_y = 1.03),
    NULL, plot_grid(plots.effsize.list[[1]], plots.effsize.list[[2]], plots.effsize.list[[3]],
        plots.effsize.list[[4]], ncol = 1, axis = "rl", align = "v", rel_heights = c(0.8,
            0.8, 0.8, 0.91), labels = c("b)", "d)", "f)", "h)"), label_size = 9,
        label_x = -0.01, label_y = 1.03, hjust = 1), ncol = 3, axis = "l", align = "v",
    rel_widths = c(0.9, -0.01, 0.6)), ncol = 1, rel_heights = c(0.05, 1))
dev.off()
```

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

5 Comparison C7 bulk properties and C6

5.1 Kolmogorov-Smornov test and one sample t-test

```
df.ttest.C7 <- data.frame(Comm.Prop = names(Funct.data)[c(5:8)], Normality = NA,
    t.stat = NA, p.value = NA)
```

```r
for (i in 1:4) {
    df.ttest.C7$Normality[i] <- ols_test_normality(na.omit(subset(Funct.data,
        Cycle == "C7", select = df.ttest.C7$Comm.Prop[i]))[, 1])[[1]][2] > 0.05
    C6.mean <- mean(na.omit(subset(Funct.data, Cycle == "C6", select = df.ttest.C7$Comm.Prop[i]))[,
        1])
    ttest.tmp <- t.test(na.omit(subset(Funct.data, Cycle == "C7", select = df.ttest.C7$Comm.Prop[i]))[,
        1], mu = C6.mean)
    df.ttest.C7$t.stat[i] <- as.numeric(ttest.tmp[1])
    df.ttest.C7$p.value[i] <- as.numeric(ttest.tmp[3])
}
```

Results one sample t-test community bulk properties

Table 2: One sample t-test bulk comunity properties

Variable	Assumptions	t-test	
Comm.Prop	Normality	t.stat	p.value
Abundance_106cellmL	TRUE	-13.482	0.000
Biomass_mgCL	TRUE	-1.104	0.288
ConsoCellobiose_mgCL	TRUE	-3.069	0.006
CUE	TRUE	0.346	0.734