HEB1410 Gut Microbiome and Human Health

Computation Lab Section

Yijia Liow

2023-11-21

Course project microbiome data

- Taxonomic composition
- Alpha-diversity
- Beta-diversity
- Installing `MaAsLin2` for differential abundance analysis

Course project microbiome data

```r
# Load the phyloseq object
ps_course <- readRDS("data/ps_course.rds")

# Inspecting individual phyloseq components
sample_df <- as.data.frame(sample_data(ps_course))
otu_df <- as.data.frame(otu_table(ps_course))
taxonomy_df <- as.data.frame(tax_table(ps_course))
```

Taxonomic composition at Phylum level

```r
# Taxonomic composition at Phylum level
ps_course_phylum <- tax_glom(ps_course, "Phylum")
df_course_phylum <- psmelt(ps_course_phylum)

# Calculate relative abundance
df_course_phylum <- df_course_phylum %>%
  group_by(Sample) %>%
  mutate(RelativeAbundance = Abundance / sum(Abundance))

# Plot with ggplot2
df_course_phylum %>%
  ggplot(aes(x = Sample, y = RelativeAbundance, fill = Phylum)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  labs(title = "Taxonomic Composition at Phylum Level",
       x = "Sample",
       y = "Relative Abundance") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Taxonomic composition at Genus level

```r
# Taxonomic composition at Genus level
ps_course_genus <- tax_glom(ps_course, "Genus")
df_course_genus <- psmelt(ps_course_genus)

# Calculate relative abundance
df_course_genus <- df_course_genus %>%
  group_by(Sample) %>%
  mutate(RelativeAbundance = Abundance / sum(Abundance)) %>%
  ungroup()

# Plot the data
df_course_genus %>%
  ggplot(aes(x = Sample, y = RelativeAbundance, fill = Genus)) +
  geom_bar(stat = "identity", position = "stack") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        legend.position = "bottom",
        legend.text = element_text(size = 4),  # Smaller legend text
        legend.key.size = unit(0.2, 'lines'))  # Adjust legend key size
```

Taxonomic composition at D-2 (baseline)

```r
# Subset data for D-2 timepoint
df_d2 <- df_course_genus %>%
  filter(timepoint == "D-2")

# Calculate relative abundance for the subset
df_d2 <- df_d2 %>%
  group_by(Sample) %>%
  mutate(RelativeAbundance = Abundance / sum(Abundance)) %>%
  ungroup()

# Plot the relative abundance for D-2 timepoint
df_d2 %>%
  ggplot(aes(x = Sample, y = RelativeAbundance, fill = Genus)) +
  geom_bar(stat = "identity", position = "stack") +
  facet_wrap(~treatment, scales = "free_x") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        legend.position = "bottom",
        legend.text = element_text(size = 4),
        legend.key.size = unit(0.2, 'lines')) +
  guides(fill = guide_legend(nrow = 8))
```

Top 20 most abundant genera

```r
# Subset data for D14 timepoint
df_d14 <- df_course_genus %>%
  filter(timepoint == "D14")

# Identify the top 20 genera based on total relative abundance at D14
top_genera_d14 <- df_d14 %>%
  group_by(Genus) %>%
  summarize(TotalAbundance = sum(RelativeAbundance)) %>%
  ungroup() %>%
  slice_max(TotalAbundance, n = 20) %>%
  arrange(desc(TotalAbundance)) %>%
  pull(Genus)

# Rename genera that are not in the top 20 as "Other"
df_d14$Genus <- ifelse(
  df_d14$Genus %in% top_genera_d14,
  df_d14$Genus, "Other")

# Reorder the factor levels so that "Other" is at the end
df_d14$Genus <- factor(df_d14$Genus, levels = c(top_genera_d14, "Other"))
```

Top 20 most abundant genera

```r
# Set the color palette
my_colors_otu <- scales::hue_pal()(length(top_genera_d14))
names(my_colors_otu) <- top_genera_d14
my_colors_otu["Other"] <- "gray"

# Plot with the top 20 genera and the 'Other' category
df_d14 %>%
  ggplot(aes(x = Sample, y = RelativeAbundance, fill = Genus)) +
  geom_bar(stat = "identity", position = "stack") +
  facet_wrap(~treatment, scales = "free_x") +
  theme_minimal() +
  scale_fill_manual(values = my_colors_otu,
                    breaks = c(top_genera_d14, "Other")) +
  labs(x = "Sample",
       y = "Relative Abundance") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        legend.position = "bottom",
        legend.text = element_text(size = 6),
        legend.key.size = unit(0.5, 'lines'))
```

Literature search

```r
# Tabulate the top 20 genera
df_d14 %>%
  group_by(Genus) %>%
  summarize(TotalAbundance = sum(Abundance)) %>%
  ungroup() %>%
  mutate(RelativeAbundance = TotalAbundance / sum(TotalAbundance)) %>%
  arrange(desc(RelativeAbundance))
```

Alpha-diversity calculation

```r
# Calculate alpha diversity for D-2 (baseline)
ps_d2 <- subset_samples(ps_course, timepoint == "D-2")
alpha_diversity_d2 <- estimate_richness(ps_d2, measures = "Shannon")
alpha_diversity_d2$Timepoint <- "D-2"
alpha_diversity_d2$Sample <- rownames(alpha_diversity_d2)

# Calculate alpha diversity for D14 (post-intervention)
ps_d14 <- subset_samples(ps_course, timepoint == "D14")
alpha_diversity_d14 <- estimate_richness(ps_d14, measures = "Shannon")
alpha_diversity_d14$Timepoint <- "D14"
alpha_diversity_d14$Sample <- rownames(alpha_diversity_d14)
```

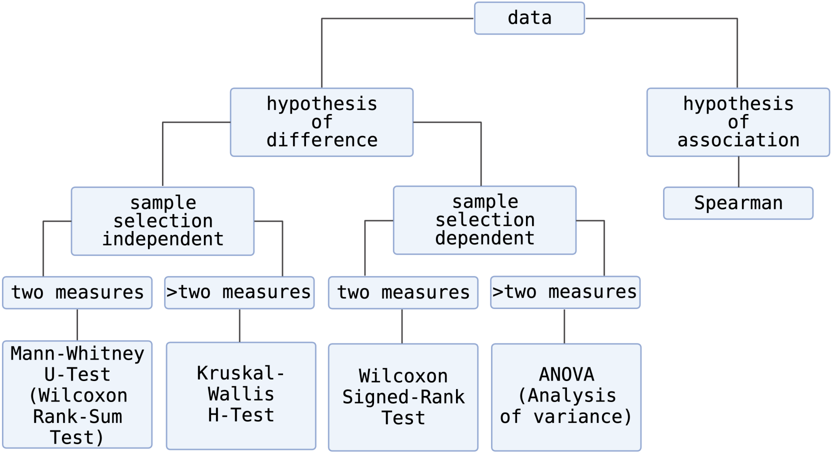
Alpha-diversity comparison

```r
# Combine both alpha-diversity dataframes
alpha_combined <- rbind(alpha_diversity_d2, alpha_diversity_d14)

# Combine and prepare metadata from different timepoints
metadata_combined <- rbind(sample_data(ps_d2), sample_data(ps_d14)) %>%
                        rownames_to_column(var = "Sample")

# Join the alpha-diversity data with the combined metadata
alpha_joined <- left_join(alpha_combined,
                          metadata_combined,
                          by = "Sample")
```

Alpha-diversity visualization

```r
# Plot alpha-diversity for both timepoints
alpha_joined %>%
  ggplot(aes(x = treatment, y = Shannon, fill = Timepoint)) +
  geom_boxplot() +
  facet_wrap(~Timepoint, scales = "free_x") +
  labs(title = "Alpha Diversity Comparison",
       x = "Treatment",
       y = "Shannon Diversity Index") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        legend.position = "right") +
  stat_compare_means(aes(group = treatment), method = "wilcox.test")
```

When to do what test?


data
hypothesis of difference
hypothesis of association
sample selection independent
sample selection dependent
Spearman
two measures
>two measures
two measures
>two measures
Mann-Whitney U-Test (Wilcoxon Rank-Sum Test)
Kruskal-Wallis H-Test
Wilcoxon Signed-Rank Test
ANOVA (Analysis of variance)

When to do what test?

1. **Parametric vs. Non-parametric**
2. **Number of Groups / Variables**
   - Two groups
   - Three or more groups
3. **Type of Data**
   - Continuous
   - Categorical (Nominal, Dichotomous, Ordinal)
4. **Purpose / Research Question**
   - Test for differences, association/correlation, prediction
5. **Dependence of Samples**
   - Example: Before-and-after measurements
6. **Tests for assumptions**
   - Normality (Shapiro-Wilk test)
   - Homogeneity of variance (Levene's test)
   - Independence of samples

Beta-diversity

```r
# Distance metric: Bray-Curtis
pcoa_bray <- ordinate(ps_course, method = "PCoA", distance = "bray")

# Prepare Bray-Curtis data for plotting
ord_bray_data <- pcoa_bray$vectors %>%
  data.frame() %>%
  mutate(Sample = rownames(.)) %>%
  left_join(sample_df %>% rownames_to_column(var = "Sample"),
    by = "Sample")

# Plotting the Bray-Curtis PCoA
bray_curtis_plot <- ord_bray_data %>%
  ggplot(aes(x = Axis.1, y = Axis.2, color = treatment)) +
  geom_point(aes(shape = timepoint)) +
  theme_minimal() +
  labs(x = "PCoA Axis 1", y = "PCoA Axis 2",
       color = "Treatment", shape = "Timepoint") +
  ggtitle("Bray-Curtis PCoA Ordination")

print(bray_curtis_plot)
```

Which axes to plot?

- **Variance explained** by each axis
- Biological or ecological interpretation/**patterns of interest**
- **Statistical testing**
- **Trial and error**

Adding percentage variation

```r
# Extract the percentage of variation explained by each axis
axis1_var <- round(pcoa_bray$values$Relative_eig[1] * 100, 2)
axis2_var <- round(pcoa_bray$values$Relative_eig[2] * 100, 2)

# Plotting the Bray-Curtis PCoA
bray_curtis_plot <- ord_bray_data %>%
  ggplot(aes(x = Axis.1, y = Axis.2, color = treatment)) +
  geom_point(aes(shape = timepoint)) +
  theme_minimal() +
  labs(x = paste("PCoA Axis 1 (", axis1_var, "%)", sep = ""),
       y = paste("PCoA Axis 2 (", axis2_var, "%)", sep = ""),
       color = "Treatment", shape = "Timepoint") +
  ggtitle("Bray-Curtis PCoA Ordination")

# Print the plot
print(bray_curtis_plot)
```

Which metric to use?

- **overall community structure**?
- diet leads to **presence/absence** of certain taxa?
- diet leads to **proliferation** of certain taxa?
- merits of incorporating **phylogenetic information**?

PERMANOVA

```
# Subsetting the data for a specific timepoint
timepoint_of_interest <- "D-2"  # Specify the timepoint
ps_subset <- subset_samples(ps_course, timepoint == timepoint_of_interest)

# Computing the distance matrix
distance_matrix <- as.matrix(distance(ps_subset, method = "bray"))

# Preparing the data for the model
data_subset <- data.frame(sample_data(ps_subset))

# Running the PERMANOVA test
adonis2(distance_matrix ~ treatment, data = data_subset)
```

Differential abundance analysis: MaAsLin2

```r
# Installing and load MaAsLin2
# Step 1: Ensure BiocManager is installed
if (!requireNamespace("BiocManager", quietly = TRUE))
    install.packages("BiocManager")

# Step 2: Install MaAsLin2 using BiocManager
BiocManager::install("Maaslin2")

# Step 3: Load MaAsLin2
library(Maaslin2)
```