

Heatmaps

Jacob Westaway

Last updated on 2021-03-26

Load packages.

```
sapply(c("tidyverse", "ggplot2", "MuMIn", "gridExtra",
        "pheatmap", "ggpubr", "DESeq2", "phyloseq"),
       require, character.only = TRUE)
```

Heatmap of parasite load (binary) and significant Genera using DESeq2 normalised data.

par_med is a binary variable that was calculated through mutating the raw ddPCR results: `mutate(par_med = ifelse(ddPCR > median(ddPCR), "High", "Low"))`

```
sample_data(ps3.Microbiome) <- sample_data(ps3.Microbiome) %>%
  unclass() %>%
  as.data.frame() %>%
  mutate(par_med = ifelse(ddPCR > median(ddPCR), "High", "Low")) %>%
  mutate(par_med = as.factor(par_med)) %>%
  mutate("Sample" = ID) %>% # change rownames to save it back into the original ps object
  mutate(Sample = paste0("AM", Sample)) %>%
  column_to_rownames("Sample")

sig_tax <- phyloseq_to_deseq2(ps3.Microbiome, ~ par_med) %>%
  calc_geo_means() %>%
  deseq_filter() %>%
  DESeq(fitType = "local", test = "Wald") %>%
  get_deseq_res_cat("par_med", "High", "Low") %>%
  select(Genus)

sig_tax <- ps3.Microbiome %>%
  tax_table() %>%
  unclass() %>%
  as.data.frame() %>%
  filter(Genus %in% sig_tax$Genus) %>%
  rownames_to_column()

phyloseq_to_deseq2(ps3.Microbiome, ~ par_med) %>%
  calc_geo_means() %>%
  deseq_filter() %>%
  DESeq(fitType = "local", test = "Wald") %>%
```

```

varianceStabilizingTransformation() %>%
assay() %>%
as.data.frame() %>% # counts of significant taxa
rownames_to_column() %>%
filter(rowname %in% sig_tax$rowname) %>%
column_to_rownames("rowname") %>%
t() %>%
as.data.frame() %>%
rownames_to_column("ID") %>%
left_join(
  (sample_data(ps3.Microbiome) %>%
    unclass() %>%
    as.data.frame() %>%
    select(ID, par_med) %>%
    mutate(ID = paste0("AM", ID))), by = "ID") %>%
column_to_rownames("ID") %>%
group_by(par_med) %>%
summarise_all(mean) %>%
column_to_rownames("par_med") %>%
t() %>%
as.data.frame() %>%
rownames_to_column() %>%
add_column( ID = c("Saliniirepens", "Hyphomonas", "Thalassobaculum", "Salinihabitans",
  "Marivivens", "Coralimargarita", "Arcobacter")) %>%
column_to_rownames("ID") %>%
select(-rowname) %>%
pheatmap(show_rownames = T, show_colnames = T, cluster_cols = F) # cluster by rows

```

Heatmap of top20 Genera and parasite load (binary).

```

top20 <- names(sort(taxa_sums(ps3.Microbiome), decreasing=TRUE))[1:20]
ps.top20 <- prune_taxa(top20, ps3.Microbiome)

sample_data(ps.top20) <- sample_data(ps.top20) %>%
  unclass() %>%
  as.data.frame() %>%
  mutate(par_med = ifelse(ddPCR > median(ddPCR), "High", "Low")) %>%
  mutate(par_med = as.factor(par_med)) %>%
  mutate("Sample" = ID) %>% # change rownames to save it back into the original ps object
  mutate(Sample = paste0("AM", Sample)) %>%
  column_to_rownames("Sample")

top20 <- phyloseq_to_deseq2(ps.top20, ~ par_med) %>%
  calc_geo_means() %>%
  DESeq(fitType = "local", test = "Wald") %>%
  varianceStabilizingTransformation() %>%
  assay() %>%
  as.data.frame() %>%
  rownames_to_column()

top20 %>%

```

```

left_join(
  ps3.Microbiome %>%
  tax_table() %>% # get the taxonomy table
  unclass() %>%
  as.data.frame() %>%
  rownames_to_column() %>%
  filter(rowname %in% top20$rowname) %>% # filter the tax table by ASV's in top20 taxa
  select(rowname, Genus) %>%
  column_to_rownames("Genus") %>% # assign the genus at the rownames instead of the ASV
  select(-rowname) %>%
  t() %>%
  as.data.frame() %>%
  rownames_to_column("ID") %>%
  left_join(
    sample_data(ps.top20) %>% # join it with the sample data
    unclass() %>%
    as.data.frame() %>%
    mutate(ID = paste0("AM", ID)) %>%
    select(par_med, ID), by = "ID" %>%
  group_by(par_med) %>%
  summarise_all(mean) %>% # get means for high/low
  column_to_rownames("par_med") %>%
  select(-ID) %>%
  t() %>%
  pheatmap(show_rownames = T, show_colnames = T, cluster_cols = F)

```

Heatmap of top20 Genera, eDNA and all environmental variables.

```

top20 %>%
  left_join(
    ps3.Microbiome %>%
    tax_table() %>% # get the taxonomy table
    unclass() %>%
    as.data.frame() %>%
    rownames_to_column() %>%
    filter(rowname %in% top20$rowname) %>% # filter tax table by ASV's that top20 taxa
    select(rowname, Genus) %>%
    column_to_rownames("Genus") %>% # assign the genus as rownames instead of ASVs
    select(-rowname) %>%
    t() %>%
    as.data.frame() %>%
    rownames_to_column("ID") %>%
    left_join(
      sample_data(ps.top20) %>% # join it with the sample data
      unclass() %>%
      as.data.frame() %>%
      mutate(ID = paste0("AM", ID)), by = "ID" %>%
    mutate(eDNA = log(ddPCR)) %>%
    select(-c(1, 22:25, 28, 30, 33:36)) %>%
    cor(method = "kendall") %>%
    round(2) %>%

```

```
as.data.frame() %>%  
select(21:26) %>%  
dplyr::slice(1:20) %>%  
pheatmap(show_rownames = T, show_colnames = T, cluster_cols = F)
```