MG ORFs R clean

Kelsey Jesser

8/5/2022

Set up environment #clear R env

```
rm(list = ls())
```

#load libraries

```
library(tidyverse)
library(tibble)
library(dplyr)
library(data.table)
library(reshape2)
library(kableExtra)
library(KEGGREST)
library(vegan)
library(ComplexHeatmap)
library(ggplot2)
library(circlize)
```

Import and format data #import annotations

```
#MicrobeAnnotator output
annot<-read.delim(file="annotations_joined.tsv", sep='\t', header=TRUE, quote="", fill=TRUE)

#protein identifiers
ids<-read.delim(file="03.Identifier_Correspondence.txt", sep='\t', header =FALSE, quote="", fill=TRUE)
colnames(ids)<-c("scaffold", "query_id")

#add identifiers to annotation df
annot<-merge(annot, ids, by="query_id")
annot<-annot %>%
  relocate ("scaffold", .after=query_id)
```

#import and filter gene abundances and metadata

```
#rel. abundance ORF matrix (already transformed by MicrobeCensus genome equivalents)
ORF_trans<-read.csv(file="matrix_all_MG_ORFs_cov_transformed.csv", header=TRUE, row.names=1)

#metadata
meta<-read.table(file="all_MG_ORFs_metadata v2.txt", sep='\t', header=TRUE)%>%
  rename("sampleID"="file_name")
```

```
meta_Inf_Case<-subset(meta, Infection_diarrhea=="Infected_Case")
meta_Inf_Control<-subset(meta, Infection_diarrhea=="Infected_Control")
meta_Uninf_Case<-subset(meta, Infection_diarrhea=="Uninfected_Case")
meta_Uninf_Control<-subset(meta, Infection_diarrhea=="Uninfected_Control")

ORF_trans<-ORF_trans%>%
  select(-c("MG_17_prodigal_MG_contigs_blastn", "MG_4_prodigal_MG_contigs_blastn", "MG_50_prodigal_MG_c
```

#format data for Kruskal-Wallis testing

```
setDT(ORF_trans, keep.rownames=TRUE)
colnames(ORF_trans)[1]<-"scaffold"

ORF_annot<-merge(annot, ORF_trans, by="scaffold")

ORF_annot_ko<- ORF_annot %>%
  subset(select = c(5, 15:117)) %>%
  group_by(ko_number) %>%
  dplyr::summarise(across(everything(), sum)) %>%
  na.omit()%>%
  column_to_rownames("ko_number")

ORF_annot_ko_t<-t(ORF_annot_ko)%>%
  as.data.frame()%>%
  rownames_to_column("sampleID")

ORF_ko_meta<-ORF_annot_ko_t %>%
  left_join(meta, by="sampleID")%>%
  select(-c(X, sampleID))
```

#run Kruskal-Wallis tests for diarrhea and DEC infection status

```
KW_raw_pvalue <- numeric(length = length(1:6221))
for (i in (1:6221)) {
  KW_raw_pvalue[i] <- kruskal.test(ORF_ko_meta[, i] ~ ORF_ko_meta$Infection_diarrhea,
    )$p.value}

#BH FDR correction
p_KW <- data.frame(
  Variable = names(ORF_ko_meta[, 1:6221]),
  KW_raw_pvalue_ko = round(KW_raw_pvalue, 10))

p_KW$BH <-
  p.adjust(p_KW$KW_raw_pvalue_ko,
    method = "BH")

#table for significant genes at corrected p < 0.05
p_KW_sig<-subset(p_KW,BH<=0.05)
colnames(p_KW_sig)[1]<-"ko"

sig_KW_ko <- subset(ORF_annot_ko, rownames(ORF_annot_ko) %in% p_KW_sig$ko) %>%
  rownames_to_column("ko")
```

#Group significant ko-annotated gene functions by pathway and create heatmaps

```
#read in ko mapping files
pathway_list <- read_tsv("path_list.txt")
pathway_ko_list <- read_tsv("path_ko.txt")

ko_list_KW<-as.data.frame(p_KW_sig$ko)
colnames(ko_list_KW)[1]<-"ko"

ko_merge_KW <- left_join(ko_list_KW, pathway_ko_list, by = "ko")
ko_merge_KW <- left_join(ko_merge_KW, pathway_list, by = "path")
ko_merge_KW <- left_join(ko_merge_KW, sig_KW_ko, by = "ko") %>%
  na.omit()
```

#calculate mean abundance of kos in each KEGG pathway

```
sig_ko_merge_KW <- ko_merge_KW %>%
  subset(select= -c(1:2)) %>%
  group_by(pathway) %>%
  dplyr::summarise(across(everything(),mean)) %>%
  drop_na()

path_remove<-read.csv("pathways_remove.csv")
sig_ko_merge_KW<-anti_join(sig_ko_merge_KW, path_remove, by = "pathway")%>%
  column_to_rownames("pathway")
```

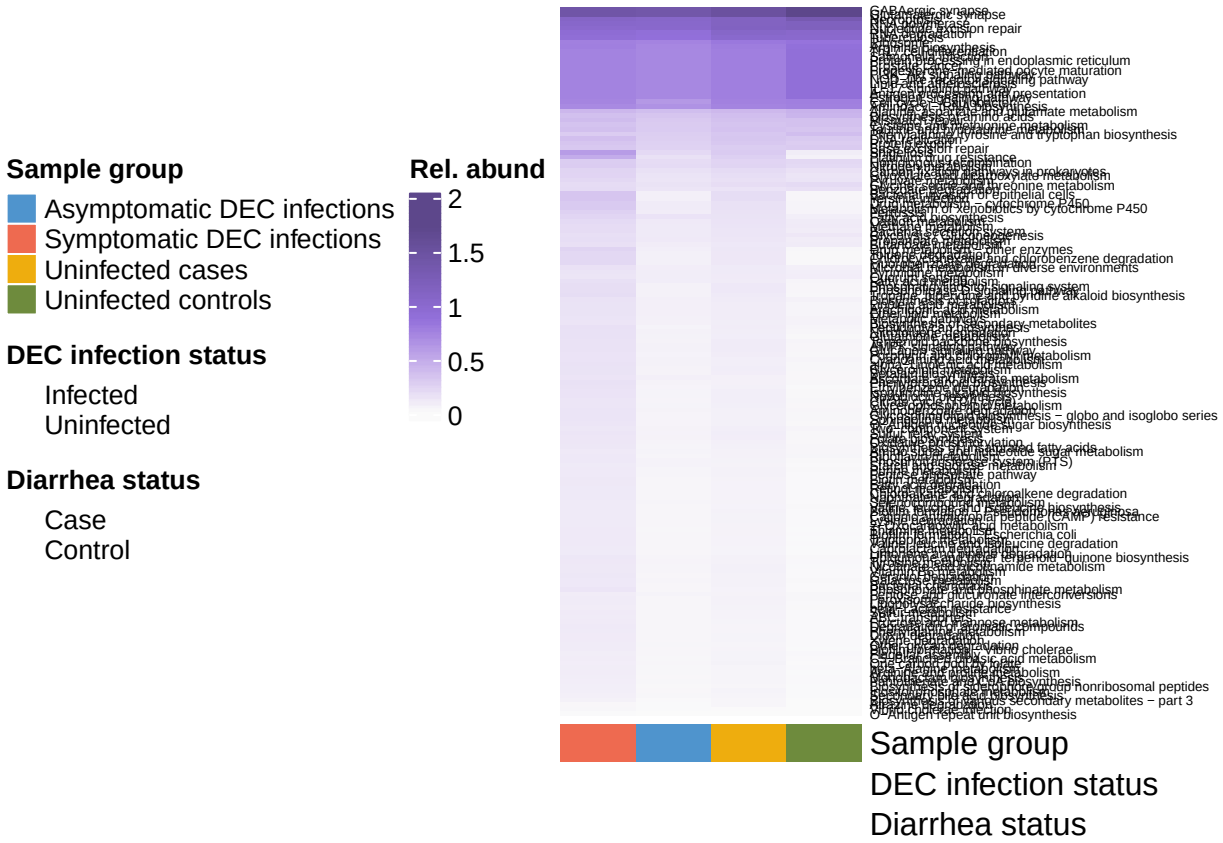
#data matrix with pathway means by DEC infection and diarrhea status

```
KW_sig_genes_inf_case<-sig_ko_merge_KW[,meta_Inf_Case$sampleID]
KW_sig_genes_inf_control<-sig_ko_merge_KW[,meta_Inf_Control$sampleID]
KW_sig_genes_uninf_case<-sig_ko_merge_KW[,meta_Uninf_Case$sampleID]
KW_sig_genes_uninf_control<-sig_ko_merge_KW[,meta_Uninf_Control$sampleID]

mean_KW_sig_genes_inf_case<-rowMeans(KW_sig_genes_inf_case)
mean_KW_sig_genes_inf_control<-rowMeans(KW_sig_genes_inf_control)
mean_KW_sig_genes_uninf_case<-rowMeans(KW_sig_genes_uninf_case)
mean_KW_sig_genes_uninf_control<-rowMeans(KW_sig_genes_uninf_control)

mean_KW<-data.frame(inf_case=round(mean_KW_sig_genes_inf_case, 5),
                                inf_control=round(mean_KW_sig_genes_inf_control, 5),
                                uninf_case=round(mean_KW_sig_genes_uninf_case, 5),
                                uninf_control=round(mean_KW_sig_genes_uninf_control, 5))
mean_KW_mat<-data.matrix(mean_KW)
```

#KW comparison mean heatmap

```
heatmap_annot<-read.csv("mean_heatmap_annot v3.csv")

col = list(Infection_diarrhea=c("Symptomatic DEC infections"="coral2", "Asymptomatic DEC infections"="s

cn=colnames(mean_KW_mat)

levels=c("Symptomatic DEC infections", "Asymptomatic DEC infections", "Uninfected cases", "Uninfected c
```

```
ha<-HeatmapAnnotation(Infection_diarrhea=heatmap_annot$Infection_diarrhea, Infection=heatmap_annot$Infe

levels=c("Symptomatic DEC infections", "Asymptomatic DEC infections", "Uninfected cases", "Uninfected c

mycols_mean <- colorRamp2(breaks = c(0,(max(mean_KW_mat)/2),max(mean_KW_mat)), colors = c("gray98", "me

h_KW<-Heatmap(mean_KW_mat,
              row_names_gp = gpar(fontsize = 5),
              show_column_names=FALSE,
              show_column_dend=FALSE,
              col=mycols_mean,
              show_row_dend = FALSE,
              row_names_max_width=max_text_width(rownames(mean_KW_mat)),
              bottom_annotation=ha,
              cluster_columns=FALSE,
              heatmap_legend_param=list(title=c("Rel. abund"), legend_height=unit(3, "cm")))
h_KW<-draw(h_KW, heatmap_legend_side="left", annotation_legend_side="left")
```



#KW heatmap with secondary KEGG pathways

```
#import mapping file
pathway_annot<-read.csv("pathway_annot.csv")
colnames(pathway_annot)[1]<-"pathway"

#merge dataframes
```

```
second_KW<-merge(ko_merge_KW, pathway_annot, by="pathway" )
second_KW <-second_KW %>% select(ko, path, Secondary, everything())

#remove duplicates
second_KW<-second_KW[!duplicated(second_KW[c(1,4)]),]

#collapse by secondary KEGG pathway
sig_ko_merge_KW_second <- second_KW %>%
  subset(select= -c(1:2,4)) %>%
  group_by(Secondary) %>%
  dplyr::summarise(across(everything(),mean)) %>%
  na.omit() %>%
  column_to_rownames("Secondary")

#subset merged data frames
KW_sig_genes_second_inf_case<-sig_ko_merge_KW_second[,meta_Inf_Case$sampleID]
KW_sig_genes_second_inf_control<-sig_ko_merge_KW_second[,meta_Inf_Control$sampleID]
KW_sig_genes_second_uninf_case<-sig_ko_merge_KW_second[,meta_Uninf_Case$sampleID]
KW_sig_genes_second_uninf_control<-sig_ko_merge_KW_second[,meta_Uninf_Control$sampleID]

#calculate means
mean_KW_sig_genes_second_inf_case<-rowMeans(KW_sig_genes_second_inf_case)
mean_KW_sig_genes_second_inf_control<-rowMeans(KW_sig_genes_second_inf_control)
mean_KW_sig_genes_second_uninf_case<-rowMeans(KW_sig_genes_second_uninf_case)
mean_KW_sig_genes_second_uninf_control<-rowMeans(KW_sig_genes_second_uninf_control)

#create data matrices
mean_KW_second<-data.frame(inf_case=round(mean_KW_sig_genes_second_inf_case, 5),
                           inf_control=round(mean_KW_sig_genes_second_inf_control, 5),
                           uninf_case=round(mean_KW_sig_genes_second_uninf_case, 5),
                           uninf_control=round(mean_KW_sig_genes_second_uninf_control, 5))
mean_KW_second<-mean_KW_second%>%
  rownames_to_column(var="second")

#remove eukaryotic and photosynthetic pathways
second_remove<-read.csv("second_remove.csv")
mean_KW_second<-anti_join(mean_KW_second, second_remove, by = "second")%>%
  column_to_rownames("second")

mean_KW_second_mat<-data.matrix(mean_KW_second)

#mean heatmap
heatmap_annot<-read.csv("mean_heatmap_annot v3.csv")

col = list(Infection_diarrhea=c("Symptomatic DEC infections"="coral2", "Asymptomatic DEC infections"="s

cn=colnames(mean_KW_mat)

levels=c("Symptomatic DEC infections", "Asymptomatic DEC infections", "Uninfected cases", "Uninfected c

ha<-HeatmapAnnotation(Infection_diarrhea=heatmap_annot$Infection_diarrhea, Infection=heatmap_annot$Infe

levels=c("Symptomatic DEC infections", "Asymptomatic DEC infections", "Uninfected cases", "Uninfected c
```

```
mycols_mean <- colorRamp2(breaks = c(0,(max(mean_KW_second_mat)/2),max(mean_KW_second_mat)), colors = c

i_KW<-Heatmap(mean_KW_second_mat,
              row_names_gp = gpar(fontsize = 10),
              show_column_names=FALSE,
              show_column_dend=FALSE,
              col=mycols_mean,
              show_row_dend = FALSE,
              cluster_columns=FALSE,
              row_names_max_width=max_text_width(rownames(mean_KW_second_mat)),
              bottom_annotation=ha,

              #rect_gp=gpar(col="white", lwd=1),
              heatmap_legend_param=list(title=c("Rel. abund"), legend_height=unit(3, "cm")))
i_KW<-draw(i_KW, heatmap_legend_side="left", annotation_legend_side="left")
```