Heatmap for relative expression of MAC101 SmT common DEGs (shared strain LU439) for Cluster 1 and 2

Julia Lienard

2024-06-01

A - Description

The heatmap was constructed for SmT DEGs of the MAC101 strain, in comparison with SmO bacteria, falling into the two main clusters of enriched GO terms (for Biological process category), identified previously during GO enrichment analysis. These DEGs are shared between the two different strains (LU439 SmT1 and/or T3 with MAC101 SmT), and had the following settings : Log2FoldChange< -1 and pvalue<0.05).

Use of following tutorial:

https://www.reneshbedre.com/blog/heatmap-with-pheatmap-package-r.html

DEGS shared by the 3 data set (LU439 T1, LU439 T3 and MAC101 T) => 137 identified.

DEGS shared by the 2 pairs (LU439 T1 or LU439 T3 with MAC101 T) => 30 +18 identified.

TOTAL = 185 DEGs upregulated for SmT samples

Input data for this analysis is the output table from the Htseq counts (matrix table) obtained during Differential gene expression analysis, done by Giulia Ribeiro.

Two main clusters of enriched GO terms were found previously for SmT samples and are:

CLUSTER 1:

- biological process involved in interaction with host
- alkane catabolic process
- phospholipid transport

CLUSTER 2:

- regulation of cell shape
- cell wall organization

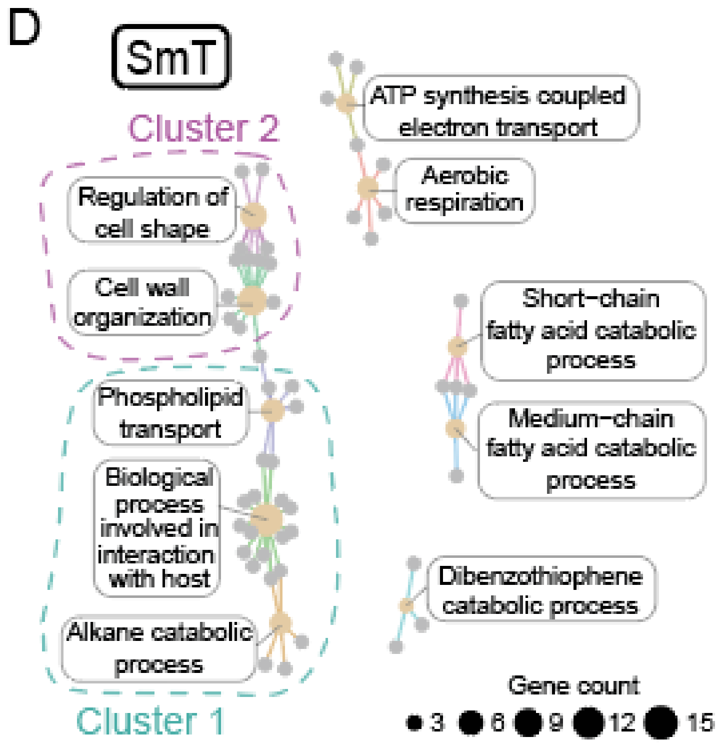
D
SmT
Cluster 2
Regulation of cell shape
Cell wall organization
Phospholipid transport
Biological process involved in interaction with host
Alkane catabolic process
Cluster 1
ATP synthesis coupled electron transport
Aerobic respiration
Short−chain fatty acid catabolic process
Medium−chain fatty acid catabolic process
Dibenzothiophene catabolic process
Gene count
● 3 ● 6 ● 9 ● 12 ● 15

Figure 1: Enrichment analysis done on the common genes, using the enricher R package from ClusterProfiler

The list of these DEGs is used to select them in the LU439 T1 vs O2 gene expression matrix, and associate them with their GO term so that only the genes under the enriched GO term are then kept for Cluster 1 and Cluster 2

Required packages

```
library(tidyverse)

library(readr)

library(dplyr)

library(pheatmap)
```

B - Loading and reformat data

1 - Loading Htseq Count - matrix table with normalized gene expression values of individual replicate samples

```
library(readr)
# loading the matrix table of gene expression analysis for MAC101 SmT vs SmO
 ↪  (centered_log2_MAC101_p6_T1REF_matrix_norm_alldata.txt)
DESeq2log2MAtrix <-
 ↪  read.delim("~/Desktop/Master/BINP39/RNAseq_visualization/MAC101/03_LU439T1_ref/01_data/0
 ↪
                               sep = "\t")

library(dplyr)
```

```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
# Removing "cds-" in front of the gene ID (pgap annotation) to homogenized
 ↪  between dataframes
# The first column with PGAP annotation IDs is named X:
DESeq2log2MAtrix$X <- gsub("cds-", "", as.character(DESeq2log2MAtrix$X))
# Renaming column names
colnames(DESeq2log2MAtrix)[1] <- "pgap_ID"

#checking the table
head(DESeq2log2MAtrix)
```

```
         pgap_ID SmO_MAC101_p6_REP1 SmO_MAC101_p6_REP2 SmO_MAC101_p6_REP3
1 pgaptmp_000001        -0.09616653         -0.7704066          0.6020316
```

```
2 pgaptmp_000002         0.48666765         -1.5815928          1.3155874
3 pgaptmp_000003        -0.05243188          0.2715122         -0.2626970
4 pgaptmp_000004        -0.10515434         -0.1141189         -0.3311051
5 pgaptmp_000005        -0.13430925         -1.4649156          0.5295264
6 pgaptmp_000006        -0.02156696         -1.3353983          0.7661285
  SmO_MAC101_p6_REP4 SmT_MAC101_p6_REP1 SmT_MAC101_p6_REP2 SmT_MAC101_p6_REP3
1         -0.4839363         0.48042124         0.13783020         0.33219703
2         -0.6652754         0.65579201        -0.01339765         0.57027170
3          0.1721121        -0.08256238        -0.07118370         0.07309162
4         -0.1125007         0.28248770         0.02734905         0.33633633
5         -1.4763963         1.48538995         0.68734556         1.25918778
6         -1.2041463         1.14129781         0.39761646         0.86625952
  SmT_MAC101_p6_REP4
1        -0.20197057
2        -0.76805284
3        -0.04784087
4         0.01670594
5        -0.88582850
6        -0.61019064
```

2 - Load genome annotation file split by GO term

a - loading and reformat

```
# Opening the formatted genome annotation file (each gene has one to several
 ↪  GO term with one GO term per line), called
 ↪  LU439_SmT_CDS_FULL_GOannot_Filt_GOsplit.txt
LU439_SmT_CDS_FULL_GOannot_split <- read.delim(

   ↪  "~/Desktop/Master/BINP39/RNAseq_visualization/LU439/1_data/1_genomeLU439T1_Blast2Goanno
                                              header = TRUE, sep = "\t")

# Checking names of columns
colnames(LU439_SmT_CDS_FULL_GOannot_split)
```

```
[1] "pgap_ID"      "product_PGAP" "GO_ID"        "GO_name"      "GO_category"
```

```
# Removing duplicates GO ID for individual gene (if the same pgap annotation
 ↪  is found to have the same GO ID, the row is removed):
library(dplyr)
LU439_SmT_CDS_FULL_GOannot_split <- LU439_SmT_CDS_FULL_GOannot_split %>%
 ↪  filter(!duplicated(cbind(pgap_ID, GO_ID)))
```

b - Filter the genes with GO terms associated with Cluster 1

```
# Filtering by selected GO ID identified by enrichment cnetplot
LU439_SmT_CDS_FULL_GOannot_split_cluster1 <- LU439_SmT_CDS_FULL_GOannot_split
 ↪  |> filter(GO_name == "biological process involved in interaction with
 ↪  host" | GO_name =="alkane catabolic process" | GO_name == "phospholipid
 ↪  transport") # 184 genes

# Remove pgap ID duplicates if any in
 ↪  LU439_SmT_CDS_FULL_GOannot_split_cluster1 :
LU439_SmT_CDS_FULL_GOannot_split_cluster1 <-
 ↪  LU439_SmT_CDS_FULL_GOannot_split_cluster1 |>
 ↪  filter(!duplicated(cbind(pgap_ID, product_PGAP)))
```

3- Load the list of common SmT DEG

```
# Load the list of common SmT DEG containing info about whether DEGs are
 ↪  shared between LU439 MAC101 T, LU439 T1 and/or T3, called
 ↪  SmT_commonDEGs_extended_withset.csv:
SmT_commonDEGs_extended <- read.delim(

   ↪  "~/Desktop/Master/BINP39/RNAseq_visualization/LU439_L2_L3_MAC101p6/REF_LU439T1genome/0

# Rename the pgap ID to homogenize with other dataframes
SmT_commonDEGs_extended$ID <- gsub("cds-", "",
 ↪  as.character(SmT_commonDEGs_extended$ID))
# Renaming column names
colnames(SmT_commonDEGs_extended)[1] <- "pgap_ID"
```

C - Select all DEGs under the GO terms identified for Cluster 1

1- Select the list of genes

```
# Merging the the annotation of Cluster 1 + the common DEG (with the sets
 ↪  info):
CommonCluster1 <- merge(LU439_SmT_CDS_FULL_GOannot_split_cluster1,
 ↪  SmT_commonDEGs_extended, by="pgap_ID") |> subset(select=c(1,2,6))

# Merging then with the matrix dataframe:
CommonMatrixCluster1 <- merge(CommonCluster1, DESeq2log2MAtrix, by="pgap_ID")

# In the Merged matrix, we pool together in a newly created column names
 ↪  "GeneProductFull" the product names with pgap_ID to obtain unique names
 ↪  as many genes have the same names and will be skip in the heatmap making
 ↪  process otherwise:
CommonMatrixCluster1$GeneProductFull <-
 ↪  paste(CommonMatrixCluster1$product_PGAP, CommonMatrixCluster1$pgap_ID,
 ↪  sep = "_")

#Extracting the information about the sets (how are shared the DEGs between
 ↪  LU439 and MAC101):
DEG_sets <- data.frame(CommonMatrixCluster1$Set)

# Keeping only the GeneProductFull and the gene expression values:
CommonMatrixCluster1_final <- CommonMatrixCluster1 |> subset(select =
 ↪  c(12,4,5,6,7,8,9,10,11))
```

2 - Make the heatmap

```
# Formating the matrix to transform the variable GeneProductFull as rownames
 ↪  instead, to be able to transform it after into a data.matrix:
rownames(CommonMatrixCluster1_final) <-
 ↪  CommonMatrixCluster1_final$GeneProductFull

# Removing now the the variable GeneProductFull:
CommonMatrixCluster1_final <- CommonMatrixCluster1_final |> subset(select =
 ↪  c(-1))
```

```
# Attributing the sets info to each DEG as it is in the matrix:
rownames(DEG_sets) <- rownames(CommonMatrixCluster1_final)
```

```
# Convert CommonMatrixCluster1_final to matrix
CommonMatrixCluster1_final_dm = data.matrix(CommonMatrixCluster1_final)
```

```
# install pheatmap
if (!require("pheatmap", quietly = TRUE))
    install.packages("pheatmap")

library(pheatmap) # version pheatmap_1.0.12


# creating heatmap
sample_group <- data.frame(sample = rep(c("SmO", "SmT"), c(4, 4)))
row.names(sample_group) <- colnames(CommonMatrixCluster1_final_dm)

heatmap_LUT1_cluster1 <-
  pheatmap(CommonMatrixCluster1_final_dm,
           annotation_col = sample_group,
           annotation_row = DEG_sets,
           cellwidth = 3,
           scale = "row",
           cellheight = 6,
           clustering_distance_cols = "manhattan",
           show_colnames = F,
           main = "MAC101 SmT vs SmO
         FC > 1 - Biological process (Cluster 1)",
          fontsize_col = 6, fontsize_row = 6)
heatmap_LUT1_cluster1
```

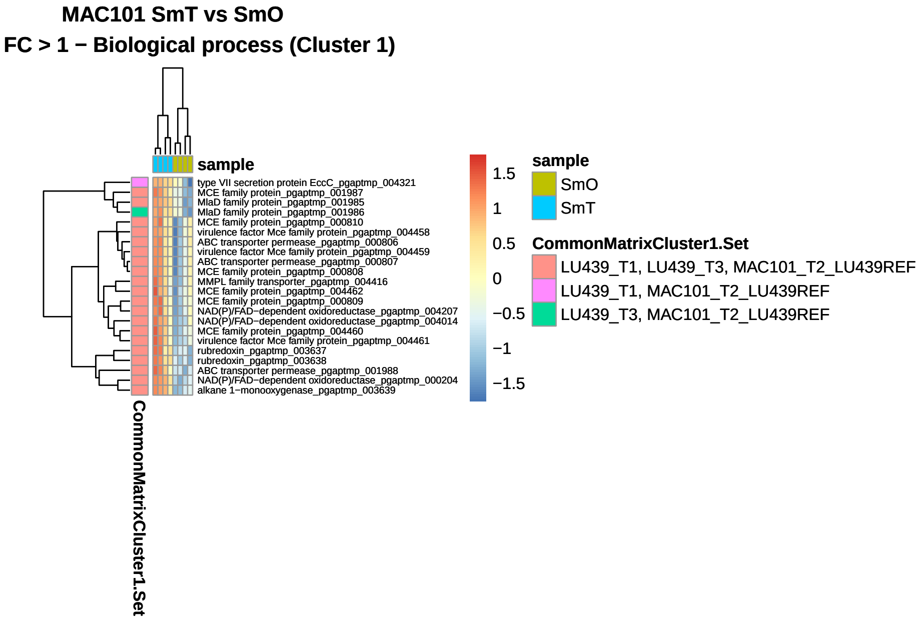

Figure 2: Heatmap for Cluster 1

```
# An R function to save pheatmap figure into pdf
# This was copied from Stackflow:
 ↪  https://stackoverflow.com/questions/43051525/how-to-draw-pheatmap-plot-to-screen-and-als


save_pheatmap_pdf <- function(x, filename, width=9, height=9) {
    stopifnot(!missing(x))
    stopifnot(!missing(filename))
    pdf(filename, width=width, height=height)
    grid::grid.newpage()
    grid::grid.draw(x$gtable)
    dev.off()
}
```

```
save_pheatmap_pdf(heatmap_LUT1_cluster1,
 ↪  "heatmap_cluster1_commonGOBP_MAC101.pdf")
```

```
pdf
  2
```

D - Select all genes under the GO terms identified for Cluster 2

1 - Select the list of genes

```
# Filtering by selected GO ID identified by enrichment cnetplot
LU439_SmT_CDS_FULL_GOannot_split_cluster2 <- LU439_SmT_CDS_FULL_GOannot_split
 ↪  |> filter(GO_name == "regulation of cell shape" | GO_name =="cell wall
 ↪  organization")

# Remove pgap ID duplicates if any in
 ↪  LU439_SmT_CDS_FULL_GOannot_split_cluster1 :
LU439_SmT_CDS_FULL_GOannot_split_cluster2 <-
 ↪  LU439_SmT_CDS_FULL_GOannot_split_cluster2 |>
 ↪  filter(!duplicated(cbind(pgap_ID, product_PGAP))) # 99 genes

# Merging the the annotation of Cluster 2 + the common DEG (with the sets
 ↪  info):
CommonCluster2 <- merge(LU439_SmT_CDS_FULL_GOannot_split_cluster2,
 ↪  SmT_commonDEGs_extended, by="pgap_ID") |> subset(select=c(1,2,6))

# Merging then with the matrix dataframe:
CommonMatrixCluster2 <- merge(CommonCluster2, DESeq2log2MAtrix, by="pgap_ID")

# In the Merged matrix, we pool together in a newly created column names
 ↪  "GeneProductFull" the product names with pgap_ID to obtain unique names
 ↪  as many genes have the same names and will be skip in the heatmap making
 ↪  process otherwise:
CommonMatrixCluster2$GeneProductFull <-
 ↪  paste(CommonMatrixCluster2$product_PGAP, CommonMatrixCluster2$pgap_ID,
 ↪  sep = "_")
write_csv(CommonMatrixCluster2, file = "CommonDEG_cluster2.csv", col_names =
 ↪  TRUE)
```

```
#Extracting the information about the sets (how are shared the DEGs between
 ↪  LU439 and MAC101):
DEG_sets_cluster2 <- data.frame(CommonMatrixCluster2$Set)

# Keeping only the GeneProductFull and the gene expression values:
CommonMatrixCluster2_final <- CommonMatrixCluster2 |> subset(select =
 ↪  c(12,4,5,6,7,8,9,10,11))
```

2 - Make the heatmap

```
# Formating the matrix to transform the variable GeneProductFull as rownames
 ↪  instead, to be able to transform it after into a data.matrix:
rownames(CommonMatrixCluster2_final) <-
 ↪  CommonMatrixCluster2_final$GeneProductFull

# Removing now the the variable GeneProductFull:
CommonMatrixCluster2_final <- CommonMatrixCluster2_final |> subset(select =
 ↪  c(-1))

# Attributing the sets info to each DEG as it is in the matrix:
rownames(DEG_sets_cluster2) <- rownames(CommonMatrixCluster2_final)
```

```
# Convert CommonMatrixCluster1_final to matrix
CommonMatrixCluster2_final_dm = data.matrix(CommonMatrixCluster2_final)
```

```
# creating heatmap
sample_group_cluster2 <- data.frame(sample = rep(c("Sm0", "SmT"), c(4, 4)))
row.names(sample_group_cluster2) <- colnames(CommonMatrixCluster2_final_dm)

heatmap_MAC101_cluster2 <-
  pheatmap(CommonMatrixCluster2_final_dm,
           annotation_col = sample_group_cluster2,
           annotation_row = DEG_sets_cluster2,
           scale = "row",
           cellwidth = 3,
           cellheight = 6,
           clustering_distance_cols = "manhattan",
           show_colnames = F,
           main = "MAC101 SmT vs SmO
         FC > 1 - Biological process (Cluster 2)",
          fontsize_col = 6, fontsize_row = 6)
heatmap_MAC101_cluster2
```

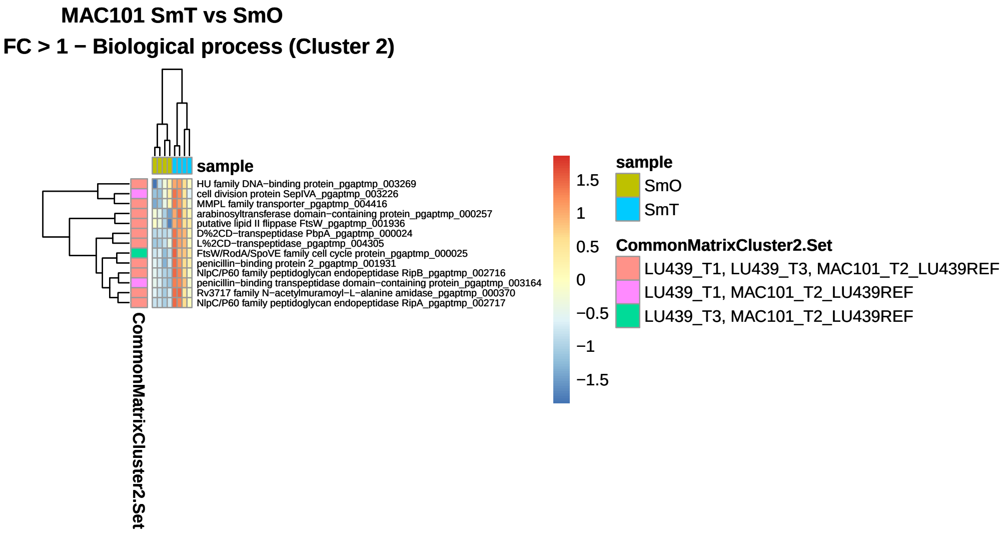

Figure 3: Heatmap for Cluster 2

```
# An R function to save pheatmap figure into pdf
# This was copied from Stackflow:
 ↪  https://stackoverflow.com/questions/43051525/how-to-draw-pheatmap-plot-to-screen-and-als


save_pheatmap_pdf <- function(x, filename, width=9, height=9) {
    stopifnot(!missing(x))
    stopifnot(!missing(filename))
    pdf(filename, width=width, height=height)
    grid::grid.newpage()
    grid::grid.draw(x$gtable)
    dev.off()
}

save_pheatmap_pdf(heatmap_MAC101_cluster2,
 ↪   "heatmap_cluster2_commonGOBP_MAC101.pdf")
```

```
pdf
  2
```