Bacterial enrichment analysis tutorial

Ali Youncha

2024-06-27

Introduction

This tutorial demonstrates how to perform taxa enrichment analysis to retrieve metabolic pathways that might be slightly enriched amongst molecular subtypes of PDAC.

First, we install and load the required libraries.

```
knitr::opts_chunk$set(echo = TRUE)
library(grid)
library(gridExtra)
library(ggplot2)
library(openxlsx)
library(dplyr)
library(stringr)
library(readxl)
library(DESeq2)
```

Make sure to install qpathway package from GenehetX github in case that you don't have it by using the following commands:

```
install.packages("BiocManager")
```

```
BiocManager::install(c("devtools","fgsea"))
```

```
devtools::install_github("GeNeHetX/qpathway")
```

Then load the libraries qpathway and fgsea and the metabolic pathway used for the taxon setenrichment analysis

```
library(qpathway)
library(fgsea)

PATH <- qpathway::loadPath("basic")
SIG <- CancerRNASig::signatures$geneset
pathways = c(PATH, SIG)
```

The first step requires loading the genus taxa metabolic pathway file that will used to perform the Taxon Set Enrichment Analysis test.

```
metabolic_data <- read.delim("C:/Users/inserm/Documents/Ali/metabolic_pathways.csv",
  header = 1, sep = ',')
```

```
taxon_list <- strsplit(as.character(metabolic_data$Taxons), ", ")
names(taxon_list) <- metabolic_data$Metabolic.Pathway
## Here, the idea is to transform this database to a large list, where each
## metabolic pathway will contain a list of genus taxa that are associated to it.
```

Then, we will load the PDAC samples file and the genus abundance file.

```
  panc_samples <- read_excel("C:/Users/inserm/Documents/panc_samples.xlsx")
  corr_patient <- panc_samples[, c("ID_Nucleic_Acid", "Idpatient", "Purist", "sous_type_visuel")]
  corr_patient <- na.omit(corr_patient)

  rawct_p3537 = read.delim("C:/Users/inserm/Documents/Ali/abundance_genus.csv",
  sep = ';', row.names = 1)
  colnames(rawct_p3537) <- gsub("_R_filtered", "", colnames(rawct_p3537))

  rownames(corr_patient) <- corr_patient$ID_Nucleic_Acid
  target = data.frame(corr_patient[colnames(rawct_p3537),])
  target = target[which(!is.na(target$Purist)),]
  rownames(target) <- target$ID_Nucleic_Acid
  rawct = rawct_p3537[,rownames(target)]
```

Once the files are loaded, the idea is to match the number of samples on both samples and genus abundance files:

```
rownames(corr_patient) <- corr_patient$ID_Nucleic_Acid
## Transform row names to samples name
target = data.frame(corr_patient[colnames(rawct_p3537),]) ## Match between the
## rows of the samples file and the column names of the genus abundance file
target = target[which(!is.na(target$Purist)),] ## Remove NAs from Purist column
rownames(target) <- target$ID_Nucleic_Acid ## Transform row names to samples names
rawct = rawct_p3537[,rownames(target)] ## selecting the columns of rawct_p3537
## whose names match the row names of the target data frame .
```

Now that we have all what we need to run a differential taxon enrichment analysis, the script containing this function is called:

```
source('C:/Users/inserm/Documents/Ali/diversity_analysis/DGE_functions.R')
```

The doDGE function requires as an input a table of genus taxa abundance and the label of interest.

```
## The label is extracted from the samples files already processed (target)

DGE_taxons <-  doDGE(rawct, target['Purist'])
```

```
## converting counts to integer mode

## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates
```

```
## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing

## -- replacing outliers and refitting for 498 genes
## -- DESeq argument 'minReplicatesForReplace' = 7
## -- original counts are preserved in counts(dds)

## estimating dispersions

## fitting model and testing
```

```
DGE_taxons <-  as.data.frame(DGE_taxons)

head(DGE_taxons)
```

```
##                   baseMean log2FoldChange     lfcSE      stat    pvalue
## Acaryochloris   10.3207787     0.62237163 0.6327376 0.9836173 0.3253037
## Acetivibrio      0.2736498     1.17106543 2.8214383 0.4150597 0.6780982
## Acetobacter     18.8600738     0.57176445 0.4811630 1.1882968 0.2347165
## Acetobacterium   1.8003325     0.21768828 1.2744722 0.1708066 0.8643758
## Acholeplasma     0.3394615     0.43551580 2.7039748 0.1610650 0.8720422
## Achromobacter  508.1210948     0.07648842 0.1540413 0.4965449 0.6195100
##                      padj
## Acaryochloris          NA
## Acetivibrio            NA
## Acetobacter            NA
## Acetobacterium         NA
## Acholeplasma           NA
## Achromobacter  0.9278403
```

Once we have the desired file that was just showed earlier. The idea is to plot a volcano plot in order to visualize the taxons that are differentially expressed in the classical and basal subtypes.

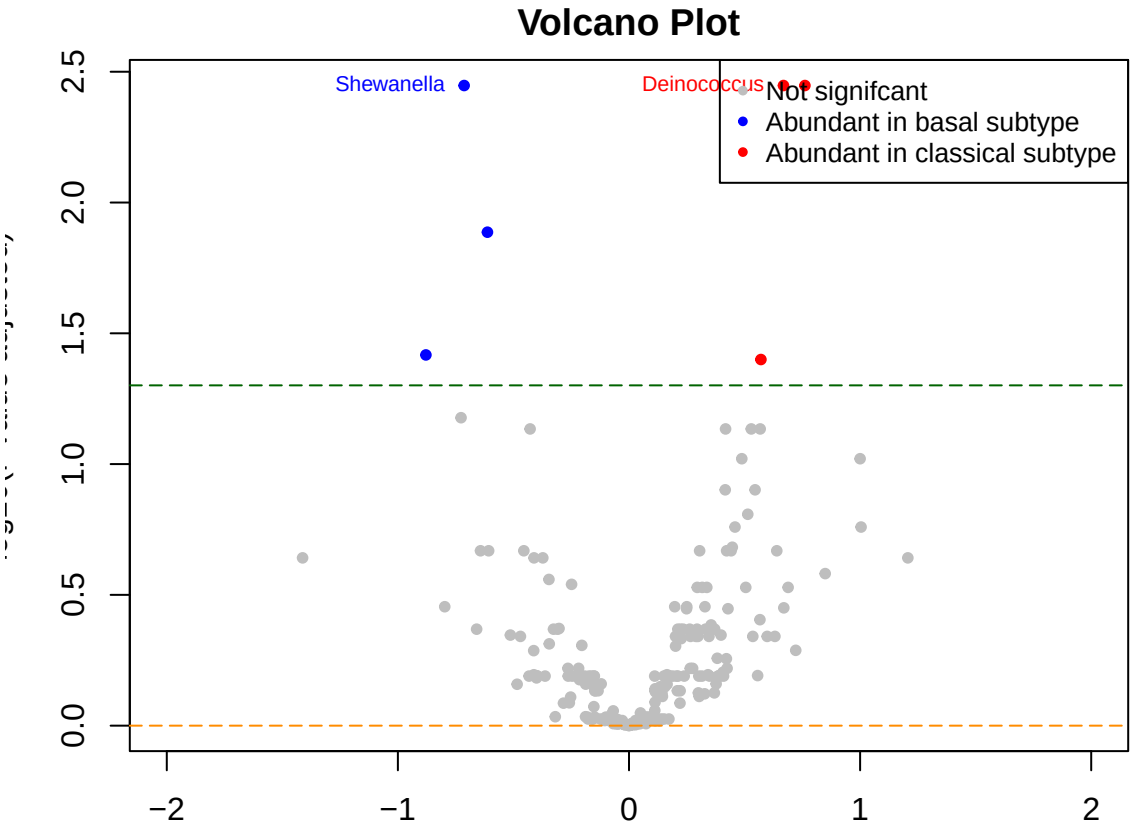
```
## Removing taxons that don't have a correspond p-value adjusted value
DGE_taxons <- DGE_taxons[complete.cases(DGE_taxons$padj), ]

## Defining the plot dimensions
par(mfrow = c(1, 1), mar = c(3.9, 3.9, 2, 2), plt = c(0.1, 0.9, 0.1, 0.9))

## Ploting the volcano plot

with(DGE_taxons, {
  logFC_range <- range(-3, 3)  # Set the range of logFC between -3 and 3
  PValue_range <- range(-log10(padj))
  plot(logFC_range, PValue_range, type = "n", main = "Volcano Plot",
       xlab = "logFC", ylab = "-log10(PValue adjusted)", xlim = c(-2, 2))
  points(log2FoldChange, -log10(padj), pch = 20,
         col = ifelse(padj <= 0.05, ifelse(log2FoldChange > 0, "red", "blue"),
```

```
                      "grey"))
  ## Abline is used to create the thresholds : pvalue = 0 and pvalue = 0.05
  abline(h = -log10(0.05), col = 'darkgreen', lty = 5)
  abline(h = 0, col = "darkorange", lty = 5)
  top_genes <- order(-log10(padj), decreasing = TRUE)[c(1,3)]
  ## Finally we add labels and legend to the plot
  text(log2FoldChange[top_genes], -log10(padj[top_genes]),
       labels = rownames(DGE_taxons)[top_genes],
       col = c("red", "blue"),
       pos = 2, cex = 0.7)
  legend("topright", legend = c("Not signifcant", "Abundant in basal subtype",
                                "Abundant in classical subtype"),
         col = c("grey", "blue", "red"), pch = 20, cex = 0.8)
})
```



The next step is to perform a taxon set enrichment analysis, for which a vector containing the statistical values of the DGE test is required. The code that follows this paragraph extracts this vector.

```
genus_stats <- data.frame(Genus = rownames(DGE_taxons), Stat = DGE_taxons$stat)

## Trasnforming the statistical value into a vector
stat <- as.vector(genus_stats$Stat)

stat <- na.omit(stat)
```

```
## Naming the rows of this vector as genus taxa.
names(stat) <- genus_stats$Genus
```

First, the script containing the required function to perform this analysis is loaded.

```
source('C:/Users/inserm/Documents/Ali/diversity_analysis/Taxon_Set_Enrichment_Analysis.R')
```

Then, we will use fgseaSimple to perform the taxon set enrichment analysis.

```
resAFU = fgseaSimple(taxon_list,stat, nperm = 1000)
resAFU = resAFU[which(as.numeric(resAFU$pval) < 0.05),]
resAFU = changeLeadingEdge(resAFU)
res_sortAFU = resAFU[order(abs(as.numeric(resAFU$NES)),decreasing=TRUE),]
write.table(res_sortAFU, "gsea_genus_pathways.tsv", sep='\t', quote=FALSE,
row.names = FALSE)
```

If you have several metabolic pathways that are enriched and you want to choose a set of metabolic pathways, adjust the following code as needed:

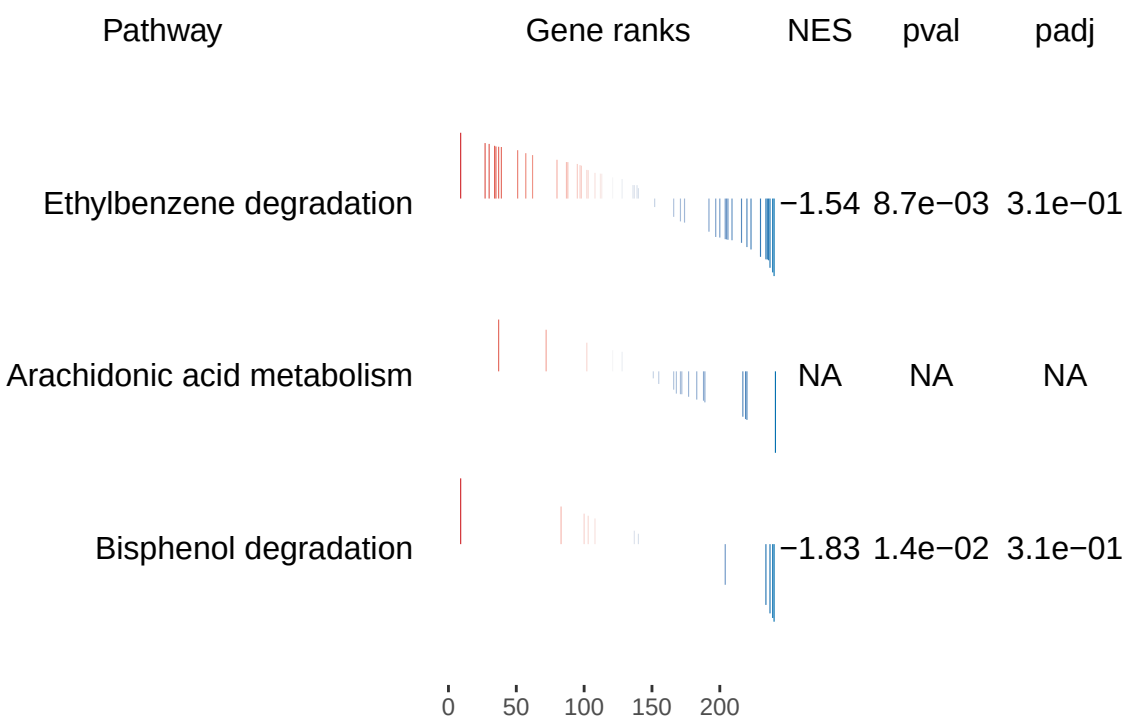
```
names_to_keep <- c( ## Metabolic pathways to keep
  "Bisphenol degradation",
  "Ethylbenzene degradation",
  "Arachidonic acid metabolism"
)

filtered_variable <- taxon_list[names(taxon_list) %in% names_to_keep]
## Creating a variable to keep only the desired metabolic pathways
```

Finally, plotting the taxon set enrichment analysis results using the qGseaTable function.

```
qGseaTable(filtered_variable, stat, res_sortAFU,gseaParam = 0.5,
colwidths = c(5,3, 0.8, 1.2, 1.2),rename=NULL )
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```


Pathway
Gene ranks
NES
pval
padj
Ethylbenzene degradation
−1.54
8.7e−03
3.1e−01
Arachidonic acid metabolism
NA
NA
NA
Bisphenol degradation
−1.83
1.4e−02
3.1e−01
0
50
100
150
200