PEC 1 Análisis de datos ómicos

Carmen Alhena Reyes Ruiz

Noviembre 2024

Índice

1. Introducción y objetivos

Los datos recogidos y analizados para la presentación de esta PEC tienen la finalidad de representar de una forma simplificada y ordenada el proceso de análisis de metabilómica haciendo uso de herramientas bioinformáticas como Rstudio, Bioconductor y Github.

2. Material y métodos

Los datos de metabolómica con los que se han trabajado han sido seleccionados del repositorio del profesor de la asignatura, disponible en:

https://github.com/nutrimetabolomics/metaboData/

Tras consultar el catálogo de bases de datos disponibles he decidido trabajar con la siguiente base de datos: 2023-UGrX-4MetaboAnalystTutorial

En el documento description.md de github puedo ver una pequeña despcripción de la base de datos, la cual fue utilizada por Sara Herraiz en un tutorial sobre análisis de datos de metabolómica utilizando MetaboAnalyst. Los datos se han descargado de Metabolomics Workbench, y su ID se encuentra en el archivo (ST000002).

Enlace de WorkBench del estudio ST000002, Intestinal Samples II pre/post transplantation:

https://www.metabolomicsworkbench.org/data/DRCCMetadata.php?Mode=Study&StudyID=ST000002&StudyType=MS&ResultType=1

En la información del proyecto “Intestinal Samples II pre/post transplantation” (ID: PR000002, DOI: 10.21228/M8WC7D) podemos ver que esta base de datos recoge datos de expresión de diferentes metabolitos en tejido intestinal de pacientes antes y después de recibir un trasplante. Los datos han sido recolectados y analizados por el laboratorio de Oliver Fiehn en el Davis Genome Center de la Universidad de California, Davis. Este estudio ha medido la respuesta metabólica de las muestras de tejido intestinal utilizando cromatografía de gases acoplada a espectrometría de masas (GC-MS) para la detección y cuantificación de los metabolitos presentes. Se centra en estudiar los cambios en el perfil metabólico de tejido intestinal humano en condiciones de pre y post trasplante.

Para este análisis se ha utilizado la versión 4.4.2 de R y la 3.20 de BiocManager.

3. Análisis y resultados

3.1 Descarga y tratamiento de los datos

El archivo de despcripción informa sobre la estuctura de los datos subidos a github y explica que los datos pueden extraerse manualmente del archivo .txt con ciertas modificaciones manuales que ya están hechas en el documento cleaned en el repositorio, tanto en txt como en csv.

En el siguiente código de R se expone la manera en la que se cargado el archivo txt modificado y los pasos de procesamiento que se ha seguido para poder crear un contenedor del tipo SummarizedExperiment. Se ha utilizado la función SummarizedExperiment del paquete metabolomicsWorkbenchR. En este caso el documento con el que estoy trabajando no tiene los metadatos porque fueron eliminados en el pre-procesamiento. Se pueden consultar tanto en la web del proyecto en Workbench como en el txt original en el repositorio de github.

```
# Bioconductor y SummarizedExperiment
if (!require("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
BiocManager::install(version = "3.20")

## Bioconductor version 3.20 (BiocManager 1.30.25), R 4.4.2 (2024-10-31
ucrt)

library(SummarizedExperiment)

#Cargar el los datos en txt
library(readr)
data <- read.table("ST000002_AN000002_clean.txt", sep = "\t", header =
FALSE, stringsAsFactors = FALSE)
```

```r
# Extraer y preparar los datos
sample_names <- as.character(data[1, -1])  # Crea un vector con nombres de las muestras de la fila 1
groups <- as.character(data[2, -1])        # Crea un vector con nombres de los grupos de la fila 2
row_names <- data[-c(1,2), 1]              # Almacenar los nombres de los metabolitos, primera columna
counts <- data[-c(1,2), -1]                # Elimina las dos primeras filas y la primera columna para quedarse solo con los datos
counts <- apply(counts, 2, as.numeric)     # Convertir los datos a formato numérico
rownames(counts) <- row_names              # Asigna los nombres de las filas a la matriz de conteos
colnames(counts) <- sample_names           # Asigna los nombres de las columnas a la matriz de conteos

# Crear el DataFrame de metadatos para las muestras
col_data <- data.frame(Group = groups, row.names = sample_names)

# Crear el objeto SummarizedExperiment (se) con el DataFrame y los datos numéricos
library(SummarizedExperiment)
se <- SummarizedExperiment(
  assays = list(counts = counts),
  colData = col_data
)



#Exploarar la clase de datos
se
```

```
## class: SummarizedExperiment
## dim: 142 12
## metadata(0):
## assays(1): counts
## rownames(142): 1-monoolein 1-monostearin ... xanthine xylose
## rowData names(0):
## colnames(12): A_684508 A_684512 ... B_684499 B_684503
## colData names(1): Group
```

```r
colData(se)
```

```
## DataFrame with 12 rows and 1 column
##                Group
##          <character>
## A_684508       After
## A_684512       After
## A_684516       After
## A_684520       After
```

```
## A_684524       After
## ...              ...
## B_684487      Before
## B_684491      Before
## B_684495      Before
## B_684499      Before
## B_684503      Before
```

```
se$Group
```

```
##  [1] "After"  "After"  "After"  "After"  "After"  "After"  "Before"
"Before"
##  [9] "Before" "Before" "Before" "Before"
```

class: SummarizedExperiment dim: 142 12 metadata(0): assays(1): counts rownames(142): 1-monoolein 1-monostearin ... xanthine xylose rowData names(0): colnames(12): A_684508 A_684512 ... B_684499 B_684503 colData names(1): Group

Esta salida nos da varios parámetros de información. En en primer lugar, expone que el objeto creado es de la clase SummarizedExperiment. Muestra las dimensiones del objeto, donde hay 142 filas que representan los compuestos químicos (rownames) y 12 columnas que representan las muestras clínicas de los pacientes (colnames). También informa de que los datos de la matriz numéricos los coge de counts (datos numericos) y de que no hay metadatos, que esto lo sabemos porque no estaban presentes en el archivo .txt modificad que se ha utilizado.

`colData(se)` contiene la información del diseño experimental, los nombres de las muestras y el grupo al que pertenece cada muestra auqnueya lo indique cada muestra en su nombre. Los niveles de Grupo son: After (post-trasplante) y Before (pre-trasplante)

`assay(se)` es la matriz principal de datos , nos da los valores de cada metabolito en cada muestra.

`rownames` indica que hay nombres de fila (en total, 142), representando compuestos o características medidas en el experimento.

En resumen, estamos trabajando con los datos de 12 muestras de pacientes (6 pre-trasplante y 6 post-trasplante) de las que existen mediciones numéricas para 142 metabolitos.

Tal y como se presentan los datos podemos analizar si hay diferencias en la expresión metabólica entre el grupo pre y el post trasplante.

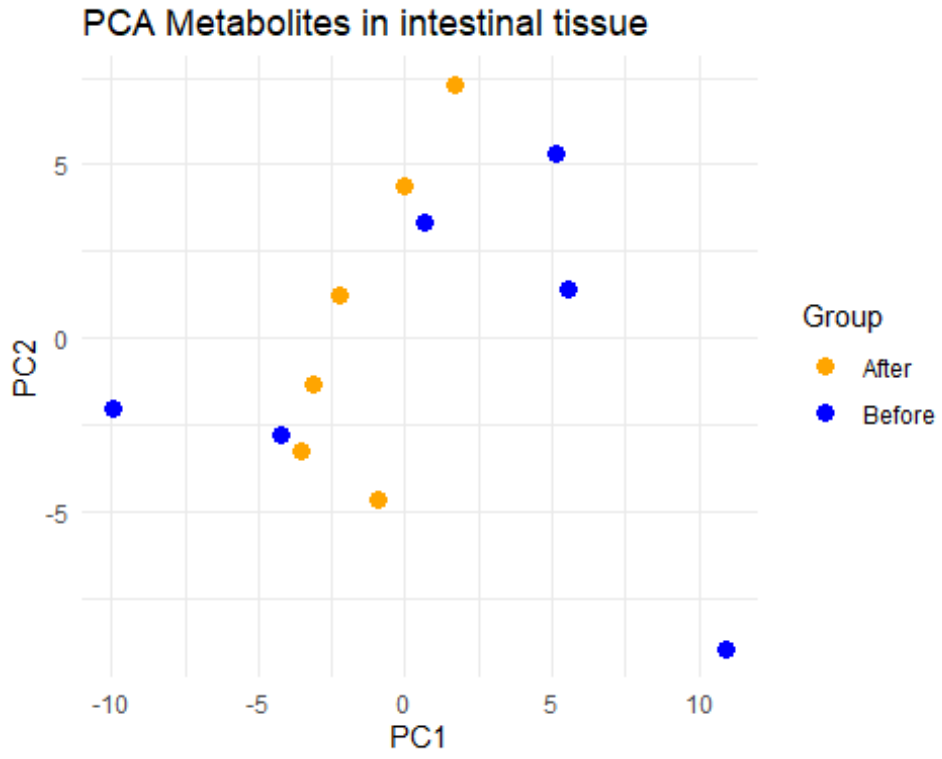
3.2 Análisis de componentes principales

```
library(ggplot2)

# PCA y dataframe de resultados
pca <- prcomp(t(assay(se)), scale. = TRUE)
```

```r
pca_df <- as.data.frame(pca$x)
pca_df$Group <- colData(se)$Group

# Gráfico de PCA
ggplot(pca_df, aes(x = PC1, y = PC2, color = Group)) +
  geom_point(size = 3) +
  theme_minimal() +
  labs(title = "PCA Metabolites in intestinal tissue",
       x = "PC1", y = "PC2") +
  scale_color_manual(values = c("Before" = "blue", "After" = "orange"))
```



En este gráfico, los puntos de ambos grupos (Before y After) están mezclados, lo que sugiere que no hay una separación clara entre los perfiles de metabolitos de los pacientes antes y después del trasplante, es decir, no hay una diferencia globaly destacada entre el estado pre y post trasplante en estos datos.

3.3. Análisis diferencial

Con los datos organizados en un ExpressionSet, se puede usar el paquete limma para realizar análisis de diferencias en los niveles de expresión de los metabolitos entre los grupos Before y After.

```r
# Análisis diferencial

# Instalar y cargar limma
if (!requireNamespace("limma", quietly = TRUE)) {
```

```r
  install.packages("BiocManager")
  BiocManager::install("limma")
}
library(limma)

##
## Adjuntando el paquete: 'limma'

## The following object is masked from 'package:BiocGenerics':
##
##     plotMA

# Crear la matriz (m1) y ajuste (fit)
m1 <- model.matrix(~ colData(se)$Group)
fit <- lmFit(assay(se), m1 )
fit <- eBayes(fit)
#Resultados significativos
results <- topTable(fit, coef = 2)
print(results)

##                          logFC    AveExpr         t     P.Value
adj.P.Val
## pyruvate              10111.17    6919.25  3.763685 0.003399353
0.4396568
## methanolphosphate  -363437.67  214949.67 -3.412412 0.006192349
0.4396568
## phosphoric acid   -4400127.50 2741197.42 -2.530515 0.028853594
0.6071504
## glutamic acid       306578.67  301135.17  2.437959 0.033911682
0.6071504
## erythritol            6492.50    4935.75  2.436171 0.034017495
0.6071504
## levanbiose           13322.50    7162.25  2.394787 0.036558667
0.6071504
## behenic acid         16019.33   22041.00  2.241560 0.047675508
0.6071504
## uracil              -53395.50   43356.75 -2.207475 0.050559555
0.6071504
## cholesterol        -637235.17  475393.08 -2.181433 0.052875140
0.6071504
## lactic acid         142384.00   84013.17  2.079465 0.062953704
0.6071504
##                            B
## pyruvate          -4.595115
## methanolphosphate -4.595115
## phosphoric acid   -4.595117
## glutamic acid     -4.595117
## erythritol        -4.595117
## levanbiose        -4.595117
## behenic acid      -4.595117
## uracil            -4.595117
```

```
## cholesterol       -4.595117
## lactic acid       -4.595118
```

En este análisis, los valores de p-value son todos mayores de 0.05, lo que indica que, tras corregir por comparaciones múltiples, no hay diferencias estadísticamente significativas en la expresión de los metabolitos antes y después del trasplante. Algunos metabolitos como “pyruvate” y “methanolphosphate” muestran una diferencia pero estas diferencias no son significativas a nivel estadístico tras el ajuste.

4. Conclusiones y subida de datos a github

Basado en estos resultados, se puede concluir que no se han encontrado cambios significativos en los niveles de los metabolitos entre los dos grupos. Esto se puede exgrapolar a la conclusión de que el tratamiento (trasplante, en este caso) no produce cambios detectables en los niveles de expresión de los metabolitos estudiados. No obstante, las limitaciones relacionadas con el tamaño de la muestra, la variabilidad biológica, y la corrección por comparaciones múltiples pueden estar impidiendo la detección de efectos reales. Un estudio con un mayor número de pacientes y un diseño experimental más detallado sería necesario para obtener conclusiones sobre el impacto del trasplante en el perfil metabólico de los pacientes.

Para subir los archivos a GithHub, he creado una cuenta en GitHub y he creado un repositorio público con el siguiente enlace:

https://github.com/careru/Reyes-Ruiz-CarmenAlhena-PEC1