Análisis de datos ómicos

PEC1

Laura Ruiz Torres

2025-04-02

Tabla de contenido

Abstract

En esta actividad se ha realizado un análisis sencillo de un dataset con datos metabolómicos de caquexia. La caquexia es un síndrome de naturaleza compleja que se caracteriza por la pérdida muscular que puede involucrar o no la pérdida de grasa. En el estudio se han analizado las muestras de orina de 77 pacientes, de los cuales 47 presentaban caquexia y 30 que no la presentaban, siendo estos últimos los pacientes control.

Objetivos

Los objetivos principales de esta actividad son:

- Conocer y gestionar datos de metabolómica usando *SummarizedExperiment()*.
- Analizar de forma general un dataset para obtener información.
- Realizar un análisis estadístico sencillo para sacar resultados relevantes del dataset como la media estadística y la desviación típica intra-grupo.

Métodos, resultados y discusión

Los apartados Métodos y Resultados se han realizado bajo un mismo epígrafe puesto que se han llevado a cabo usando R y resulta más sencillo de seguir si comandos y resultados van seguidos.

Consideraciones previas

El análisis del dataset se ha realizado en la versión de R 4.3.2.

El dataset metabolómico elegido se ha obtenido desde del enlace de GitHub proporcionado para realizar la actividad. Este contiene datos sobre metabolitos relacionados con la caquexia

Generación del objeto SummarizedExperiment

En primer lugar, se ha cargado el csv con los datos de metaboloma.

```
# Librerías
library(readr)
library(SummarizedExperiment)
```

```
## Loading required package: MatrixGenerics
```

```
## Loading required package: matrixStats
```

```
## Warning: package 'matrixStats' was built under R version 4.3.3
```

```
##
## Attaching package: 'MatrixGenerics'
```

```
## The following objects are masked from 'package:matrixStats':
##
##     colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
##     colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
##     colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
##     colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
##     colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
##     colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
##     colWeightedMeans, colWeightedMedians, colWeightedSds,
##     colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
##     rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
##     rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
##     rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
##     rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
##     rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
##     rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
##     rowWeightedSds, rowWeightedVars
```

```
## Loading required package: GenomicRanges
```

```
## Loading required package: stats4

## Loading required package: BiocGenerics

##
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs

## The following objects are masked from 'package:base':
##
##     anyDuplicated, aperm, append, as.data.frame, basename, cbind,
##     colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
##     get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply,
##     match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
##     Position, rank, rbind, Reduce, rownames, sapply, setdiff, sort,
##     table, tapply, union, unique, unsplit, which.max, which.min

## Loading required package: S4Vectors

##
## Attaching package: 'S4Vectors'

## The following object is masked from 'package:utils':
##
##     findMatches

## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname

## Loading required package: IRanges

##
## Attaching package: 'IRanges'

## The following object is masked from 'package:grDevices':
##
##     windows

## Loading required package: GenomeInfoDb

## Warning: package 'GenomeInfoDb' was built under R version 4.3.3

## Loading required package: Biobase

## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
##
## Attaching package: 'Biobase'

## The following object is masked from 'package:MatrixGenerics':
##
##     rowMedians

## The following objects are masked from 'package:matrixStats':
##
##     anyMissing, rowMedians
```

```
# Cargar el dataset
csv_cachexia <- read_csv("human_cachexia.csv")
```

```
## Rows: 77 Columns: 65

## ── Column specification ────────────────────────────────────────────────────
## Delimiter: ","
## chr  (2): Patient ID, Muscle loss
## dbl (63): 1,6-Anhydro-beta-D-glucose, 1-Methylnicotinamide, 2-Aminobutyrate,...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(csv_cachexia)
```

```
## # A tibble: 6 × 65
##   `Patient ID` `Muscle loss` `1,6-Anhydro-beta-D-glucose` `1-Methylnicotinamide`
##   <chr>        <chr>                                <dbl>                   <dbl>
## 1 PIF_178      cachexic                              40.8                    65.4
## 2 PIF_087      cachexic                              62.2                   340.
## 3 PIF_090      cachexic                             270.                     64.7
## 4 NETL_005_V1  cachexic                             154.                     53.0
## 5 PIF_115      cachexic                              22.2                    73.7
## 6 PIF_110      cachexic                             213.                     31.8
## # ℹ 61 more variables: `2-Aminobutyrate` <dbl>, `2-Hydroxyisobutyrate` <dbl>,
## #   `2-Oxoglutarate` <dbl>, `3-Aminoisobutyrate` <dbl>,
## #   `3-Hydroxybutyrate` <dbl>, `3-Hydroxyisovalerate` <dbl>,
## #   `3-Indoxylsulfate` <dbl>, `4-Hydroxyphenylacetate` <dbl>, Acetate <dbl>,
## #   Acetone <dbl>, Adipate <dbl>, Alanine <dbl>, Asparagine <dbl>,
## #   Betaine <dbl>, Carnitine <dbl>, Citrate <dbl>, Creatine <dbl>,
## #   Creatinine <dbl>, Dimethylamine <dbl>, Ethanolamine <dbl>, Formate <dbl>, …
```

Con solo cargar el dataset ya tenemos una idea de los datos que contiene, hay 77 filas y 65 columnas diferentes. La primera columna “Patient ID” corresponde al código de la muestra, y la segunda columna “Muscle loss” al estado del paciente, es decir si presenta caquexia o no. A los que no presentan caquexia, los consideramos individuos control. El resto de columnas son todos los valores numéricos de los metabolitos presentes y estudiados.

Estos metabolitos van a dar lugar a una matriz. Al estudiar la información referente al paquete, se ha podido observar que, los pacientes deben ir en columnas y los metabolitos en las filas, al contrario de como tenemos el dataset cargado inicialmente. Para solucionar este problema se traspone la matriz.

```r
# Matriz
matriz <- t(as.matrix(csv_cachexia[, -c(1, 2)])) # se le da la vuelta a
las filas y a las columnas
rownames(matriz) <- colnames(csv_cachexia[, -c(1, 2)]) # metabolitos
colnames(matriz) <- csv_cachexia$`Patient ID` # pacientes
```

Para continuar con los pasos a seguir, se genera un dataframe con el código de los pacientes y a qué grupo pertenecen, si a caquéxia o a control. A este dataframe le vamos a dar el nombre col_data por mantener, de alguna manera, la coherencia con la nomenclatura que se suele utilizar y para que sea más intuitivo construir el SummarizedExperiment posteriormente.

```r
# Dataframe
col_data <- DataFrame(
  Patient_ID = csv_cachexia$`Patient ID`,
  Muscle_loss = csv_cachexia$`Muscle loss`
)
rownames(col_data) <- col_data$Patient_ID # asignar el nombre a las filas
```

Se genera otro dataframe con la información referente a los metabolitos.

```r
# Dataframe
row_data <- DataFrame(
  Metabolite = rownames(matriz)
)
rownames(row_data) <- row_data$Metabolite
```

En este paso se construye el objeto SummarizedExperiment con la función del mismo nombre

```r
sumexp <- SummarizedExperiment(
  assays = list(Expresion = matriz),
  colData = DataFrame(col_data),
  rowData = DataFrame(row_data)
)

# Comprobaciones
sumexp
```

```
## class: SummarizedExperiment
## dim: 63 77
## metadata(0):
## assays(1): Expresion
## rownames(63): 1,6-Anhydro-beta-D-glucose 1-Methylnicotinamide ...
##   pi-Methylhistidine tau-Methylhistidine
## rowData names(1): Metabolite
## colnames(77): PIF_178 PIF_087 ... NETL_003_V1 NETL_003_V2
## colData names(2): Patient_ID Muscle_loss
```

```r
colData(sumexp)
```

```
## DataFrame with 77 rows and 2 columns
##                Patient_ID Muscle_loss
##               <character> <character>
## PIF_178           PIF_178    cachexic
## PIF_087           PIF_087    cachexic
## PIF_090           PIF_090    cachexic
## NETL_005_V1   NETL_005_V1    cachexic
## PIF_115           PIF_115    cachexic
## ...                   ...         ...
## NETCR_019_V2 NETCR_019_V2     control
## NETL_012_V1   NETL_012_V1     control
## NETL_012_V2   NETL_012_V2     control
## NETL_003_V1   NETL_003_V1     control
## NETL_003_V2   NETL_003_V2     control
```

```
rowData(sumexp)
```

```
## DataFrame with 63 rows and 1 column
##                                       Metabolite
##                                      <character>
## 1,6-Anhydro-beta-D-glucose 1,6-Anhydro-beta-D-g..
## 1-Methylnicotinamide         1-Methylnicotinamide
## 2-Aminobutyrate                   2-Aminobutyrate
## 2-Hydroxyisobutyrate         2-Hydroxyisobutyrate
## 2-Oxoglutarate                     2-Oxoglutarate
## ...                                           ...
## cis-Aconitate                       cis-Aconitate
## myo-Inositol                         myo-Inositol
## trans-Aconitate                   trans-Aconitate
## pi-Methylhistidine             pi-Methylhistidine
## tau-Methylhistidine           tau-Methylhistidine
```

El objeto SummarizedExperiment tiene algunas diferencias frente a un ExpressionSet, uno de los más notables es que el primero está hecho para ómicas variadas, de metabolitos, ARN... mientras que el segundo se limita a datos de expresión génica. El acceso a los datos es diferente, mientras que en el primero se utiliza rowData() o colData(), en el segundo se usa fData() o pData().

Análisis exploratorio

El análisis exploratorio de los datos se puede realizar desde diversos puntos de vista y hay múltiples formas de representarlo.

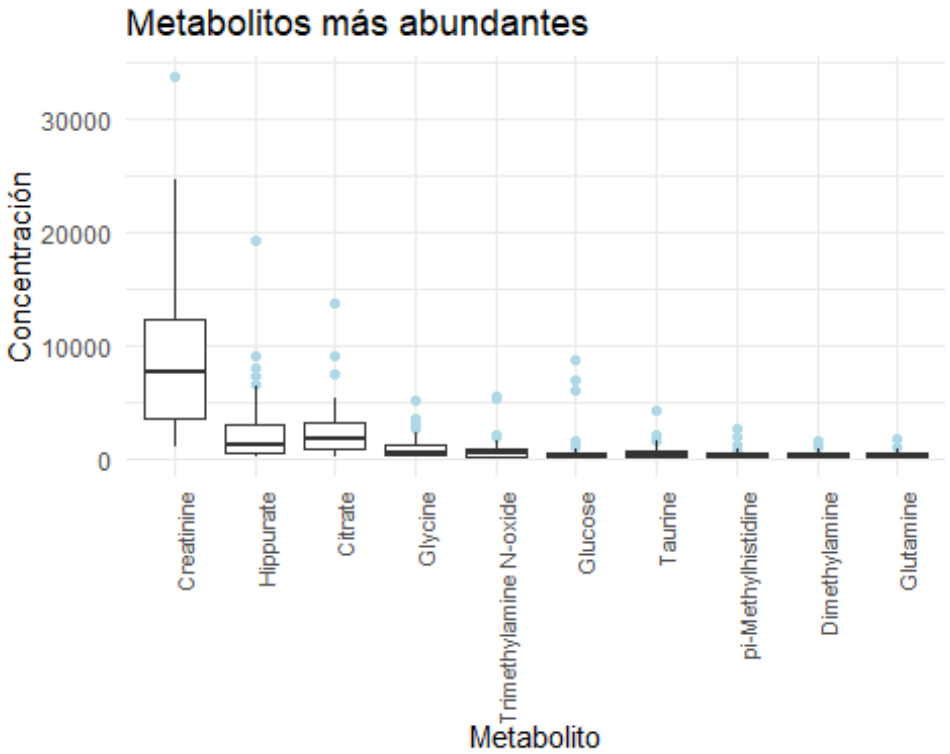
```
# dimensiones
dim(sumexp)
```

```
## [1] 63 77
```

A partir de esto sabemos que el objeto tiene 63 filas y 77 columnas. Por lo tanto el estudio ha contado 63 metabolitos y 77 pacientes.

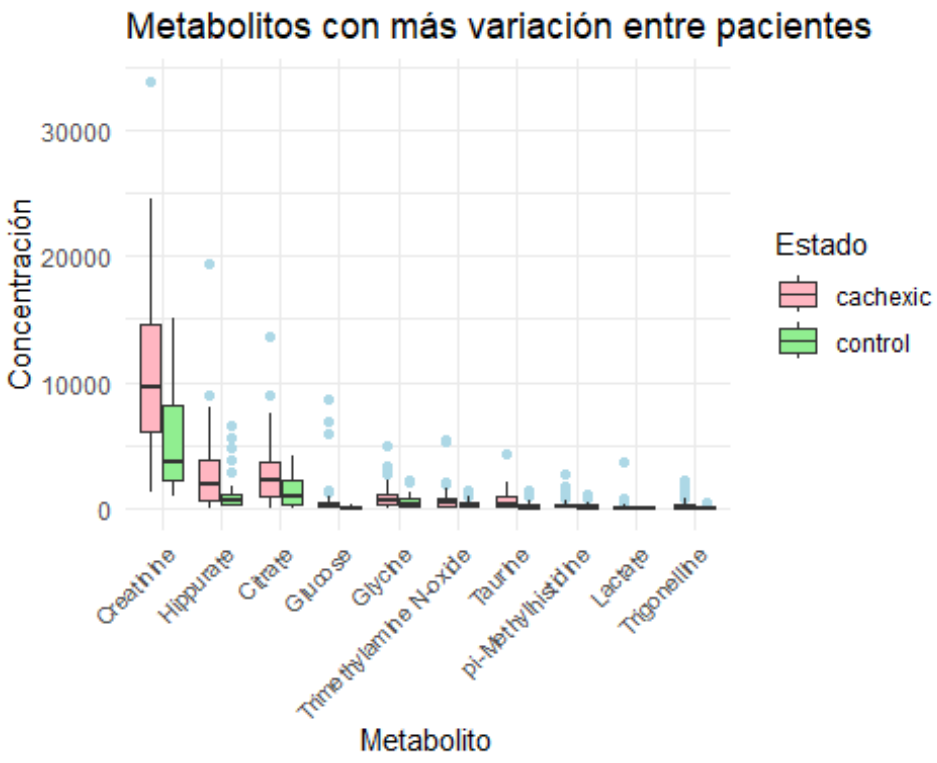
Podemos ver qué metabolitos están más representados en las muestras, y para ello se va a realizar un gráfico boxplot con los 10 más abundantes.

```
library(reshape2)
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.3.3

media_abund <- rowMeans(assay(sumexp, "Expresion")) # cálculo de la
abundancia media por meteabolito

abund_10 <- names(sort(media_abund, decreasing = TRUE))[1:10] # orden
metabolitos
matriz_10 <- assay(sumexp, "Expresion")[abund_10, ]

matriz_10_long <- melt(matriz_10)
colnames(matriz_10_long) <- c("Metabolito", "Paciente", "Concentración")

# boxplot
ggplot(matriz_10_long, aes(x = Metabolito, y = Concentración)) +
  geom_boxplot(outlier.color = "lightblue", outlier.shape = 16) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, size = 8)) +
  labs(title = "Metabolitos más abundantes",
       x = "Metabolito", y = "Concentración")
```



También resulta interesante hacer un gráfico para ver cuáles son los metabolitos que más varían entre pacientes.

```
desvest_metabolitos <- apply(assay(sumexp, "Expresion"), 1, sd)

# Ordenar los metabolitos por desviación estándar
desvest_metabolitos_10 <- names(sort(desvest_metabolitos, decreasing =
TRUE))[1:10]  # metabolitos más variables

matriz_10_desvest <- assay(sumexp, "Expresion")[desvest_metabolitos_10, ]

# Largo
matriz_10_desvest_long <- melt(matriz_10_desvest)
colnames(matriz_10_desvest_long) <- c("Metabolito", "Paciente",
"Concentración")

# caquexia/control
matriz_10_desvest_long$Estado <-
as.factor(colData(sumexp)$Muscle_loss[matriz_10_desvest_long$Paciente])

# boxplot
ggplot(matriz_10_desvest_long, aes(x = Metabolito, y = Concentración,
fill = Estado)) +
  geom_boxplot(outlier.color = "lightblue", outlier.shape = 16) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 8)) +
  labs(title = "Metabolitos con más variación entre pacientes",
       x = "Metabolito", y = "Concentración", fill = "Estado") +
  scale_fill_manual(values = c("control" = "lightgreen", "cachexic" =
"lightpink"))
```



Referencias

Para la realización de esta actividad se ha usado el manual del paquete 'SummarizedExperiment' actualizado el 28 de marzo del 2025.

El database que se ha utilizado se encuentra en este enlace https://github.com/nutrimetabolomics/metaboData/blob/main/Datasets/2024-Cachexia/description.md