PEC 1

Petrissans Moll, Sofia

2024-10-28

Contents

Resumen ejecutivo

El siguiente informe detalla el análisis de un dataset de metabolitos en pacientes que presentan caquexia y pacientes controles.

El proceso ha consitido en la elección de un dataset, la descarga de los datos y la creación de un contendeor de tipo `SummarizedExperiment`. Para la exploración y análisis del dataset se ha hecho uso de R y bibliotecas específicas para manipular y visualizar los datos. El análisis para obtener una visión general de los datos, ha consitido en la descripción de los datos y metadatos de nuestro objeto, seguido de una descripción de grupos. Un análisis de PCA para explorar la variabilidad de los datos junto a su visualización y por último, gráficos boxplot para observar la diferencia de los metabolitos medidos entre los dos grupos.

El uso de este dataset permite realizar un repaso y ampliación de los objetivos de esta primera parte de la asignatura, por lo que, los resultados obtenidos durante el análisis de los datos no pretender estudiar con detalle cada metabolito del dataset, sino aplicar las técnicas de análisis ómicos adquiridos hasta ahora.

Objetivos del estudio

El objetivo de este estudio es demostrar que se han alcanzado los objetivos y competencias planteados a lo largo del primer reto de la asignatura, mediante la realización de un análisis de un conjunto de datos, en este caso, de metabolitos en pacientes cachexicos y controles. Esto implica aplicar habilidades aprendidas en el curso.

Los objetivos específicos son:

- Generar un objeto estructurado SummarizedExperiment que facilite el manejo y análisis de los datos.
- Identificar y cuantificar diferencias en los perfiles de metabolitos entre pacientes cachexicos y controles.
- Aplicar técnicas de análisis multivariado, como PCA, para explorar patrones subyacentes en los datos, a la vez que permiten visualizaragrupaciones naturales o erróneas
- Utilizar repositorios como github, que permiten distribuir y administrar código.

Este enfoque no solo refuerza el aprendizaje de conceptos ya tratados, sino que también fomenta la adquisición de nuevas habilidades necesarias para el análisis de datos en contextos biológicos.

Materiales y Métodos

Origen y naturaleza de los datos.

El dataset analizado Los datos provienen de un paquete de R, `specmine.datasets`, pero debido a la imposibilidad de descargar este paquete al no estar diponible, se han obtenido del repositorio de github https://github.com/nutrimetabolomics/metaboData/.

Los datos descargados (human_cachexia) están disponibles en formato CSV. Los metadotos estaban incluidos en las primeras dos columnas (Patient.ID y Muscle.loss) del dataset por lo que primero se dividieron los datos que aportaban información cualitativa de los que aportaban información cuantitativa. Se emplearon métodos estadísticos y computacionales para llevar a cabo el análisis, utilizando un enfoque basado en R.

Herramientas informáticas.

Se utilizó R como lenguaje de programación principal, aprovechando bibliotecas como:

- SummarizedExperiment
- ggplot2
- tidyr
- S4Vectors

Procedimiento general de análisis.

- Análisis descriptivo de los datos. Se extrajeron los datos experimentales del objeto SummariezExperiment
- Descirpción de grupos. Se identificaro los grupos de pacientes segun el estado de caquexia.
- Análisis de componentes princiaples (PCA). Los datos fueron preparados y escalados para después aplicar la función `prcompy` obtener los principales componentes que explican la variabilidad de los datos.
- Boxplots por grupo para cada metabolito.

Resultados

Selección de un dataset de metabolómica

El dataset que he elegido proviene del repositorio de github https://github.com/nutrimetabolomics/metaboData/.

La caquexia es un complejo síndrome que causa périda de masa muscular. Normalmente está asociada a una enfermedad subyacente como puede ser el cáncer. Para el estudio se recogieron un total de 77 muestras de orina de pacientes de los cuales 47 tenian caquexia y 30 pacientes control.

```
library(SummarizedExperiment)

cachexia <- read.csv("C:/Users/sofia/Desktop/Master UOC/Tercer cuatri/Análisis datos Ómicos/PEC_1/human_

head(cachexia[, 1:15])
```

```
##    Patient.ID Muscle.loss X1.6.Anhydro.beta.D.glucose X1.Methylnicotinamide
## 1     PIF_178    cachexic                       40.85                  65.37
## 2     PIF_087    cachexic                       62.18                 340.36
## 3     PIF_090    cachexic                      270.43                  64.72
## 4 NETL_005_V1    cachexic                      154.47                  52.98
## 5     PIF_115    cachexic                       22.20                  73.70
## 6     PIF_110    cachexic                      212.72                  31.82
##   X2.Aminobutyrate X2.Hydroxyisobutyrate X2.Oxoglutarate X3.Aminoisobutyrate
## 1            18.73                 26.05           71.52              1480.30
## 2            24.29                 41.68           67.36               116.75
## 3            12.18                 65.37           23.81                14.30
## 4           172.43                 74.44         1199.91               555.57
## 5            15.64                 83.93           33.12                29.67
## 6            18.36                 80.64           47.94                17.46
##   X3.Hydroxybutyrate X3.Hydroxyisovalerate X3.Indoxylsulfate
## 1              56.83                 10.07            566.80
## 2              43.82                 79.84            368.71
## 3               5.64                 23.34            665.14
## 4             175.91                 25.03            411.58
## 5              76.71                 69.41            165.67
## 6              31.82                 35.16            183.09
##   X4.Hydroxyphenylacetate Acetate Acetone Adipate
## 1                  120.30  126.47    9.49   38.09
## 2                  432.68  212.72   11.82  327.01
## 3                  292.95  314.19    4.44  131.63
## 4                  214.86   37.34  206.44  144.03
## 5                   97.51  407.48   44.26   15.03
## 6                  132.95   81.45   14.44   25.28
```

Summarized Experiment

Una vez descargados los datos se procede a la creación de un contenedor del tipo SummarizedExperiment que contenga los datos y los metadatos

Se puede observar que cada fila representa una muestra, en este caso, un paciente. La primera columna da infromación del identificador de cada paciente, mientras que la segunda columna indica si el paciente

presenta o no caquexia, seguido de las otras 63 columnas que represetan la medición de diferentes metabolitos observados en el estudio.

Como se explica en bioconductor (https://bioconductor.org/packages/release/bioc/vignettes/SummarizedExperiment/inst/doc/SummarizedExperiment.html), `SummarizedExperimentes` es un contenedor de tipo matriz donde las filas representan características de interes (en este caso metabolitos de interes) y las columnas representan muestras.

Primero, dividiré los metadatos, aquellos datos que nos aportan información de los valores medidos, en este caso, primera y segunda columna `Patient.ID` y `Muscle.loss`. Y los datos numéricos, el resto y me aseguro que sean de tipo numérico.

Como los pacientes deben quedar representados en las columnas, hay que transponer los valores numéricos, que estaban en las columnas, para que ahora pasen a las filas.

```
library(S4Vectors)

numeric_data <- as.data.frame(lapply(cachexia[, -c(1, 2)], function(x) as.numeric(as.character(x))))

# Transponer numeric_data para que las columnas representen pacientes
numeric_data_t <- t(numeric_data)

# Crear los metadatos usando las columnas 'Patient.ID' y 'Muscle.loss'
metadata <- DataFrame(Patient.ID = cachexia$Patient.ID, Muscle.loss = cachexia$Muscle.loss)
```

Por último, creo el objeto SummarizedExperiment

```
# Crear el objeto SummarizedExperiment
se <- SummarizedExperiment(
    assays = list(counts = as.matrix(numeric_data_t)),
    colData = metadata
)
se
```

```
## class: SummarizedExperiment
## dim: 63 77
## metadata(0):
## assays(1): counts
## rownames(63): X1.6.Anhydro.beta.D.glucose X1.Methylnicotinamide ...
##   pi.Methylhistidine tau.Methylhistidine
## rowData names(0):
## colnames: NULL
## colData names(2): Patient.ID Muscle.loss
```

Podemos ver la dimensión del objeto creado que contiene 63 variables, metabolitos medidos, y 77 pacientes.

Por último, guardo el objeto:

```
save(se, file = "se_cachexia.Rda")
```

Exploración del dataset

Llevar a cabo una exploración del dataset que proporcione una visión general del mismo

Análisis descriptivo

Para recuperar los datos experimentales de un SummarizedExperimentobjeto, se puede utilizar el assays()descriptor de acceso (https://bioconductor.org/packages/release/bioc/vignettes/SummarizedExperiment/inst/doc/SummarizedExperiment.html).

```
head(assays(se)$counts[, 1:10])
```

```
##                                [,1]   [,2]   [,3]    [,4]  [,5]   [,6]   [,7]
## X1.6.Anhydro.beta.D.glucose   40.85  62.18 270.43  154.47 22.20 212.72 151.41
## X1.Methylnicotinamide         65.37 340.36  64.72   52.98 73.70  31.82  36.60
## X2.Aminobutyrate              18.73  24.29  12.18  172.43 15.64  18.36   8.67
## X2.Hydroxyisobutyrate         26.05  41.68  65.37   74.44 83.93  80.64  42.52
## X2.Oxoglutarate               71.52  67.36  23.81 1199.91 33.12  47.94 223.63
## X3.Aminoisobutyrate         1480.30 116.75  14.30  555.57 29.67  17.46  56.26
##                               [,8]  [,9]  [,10]
## X1.6.Anhydro.beta.D.glucose 31.50 51.42 117.92
## X1.Methylnicotinamide        6.82 30.27  52.46
## X2.Aminobutyrate             4.18  7.54  19.49
## X2.Hydroxyisobutyrate       12.94 34.81  72.24
## X2.Oxoglutarate             25.03 80.64  73.70
## X3.Aminoisobutyrate          8.67 17.99  57.97
```

Para acceder a los metadatos se puede usar la función `colData()`.

```
colData(se)
```

```
## DataFrame with 77 rows and 2 columns
##       Patient.ID Muscle.loss
##      <character> <character>
## 1        PIF_178    cachexic
## 2        PIF_087    cachexic
## 3        PIF_090    cachexic
## 4    NETL_005_V1    cachexic
## 5        PIF_115    cachexic
## ...          ...         ...
## 73  NETCR_019_V2     control
## 74   NETL_012_V1     control
## 75   NETL_012_V2     control
## 76   NETL_003_V1     control
## 77   NETL_003_V2     control
```

Como hemos establecido, contiene 2 columnas que descriptivas para cada fila de muestra que es la identificación de cada paciente y si presenta o no el síndrome de estudio.

Es interesante acceder al metadato `Muscle.loss` y observar los dos niveles:

```
se[, se$Muscle.loss == "cachexic"]
```

```
## class: SummarizedExperiment
## dim: 63 47
## metadata(0):
```

```
## assays(1): counts
## rownames(63): X1.6.Anhydro.beta.D.glucose X1.Methylnicotinamide ...
##   pi.Methylhistidine tau.Methylhistidine
## rowData names(0):
## colnames: NULL
## colData names(2): Patient.ID Muscle.loss
```

```
se[, se$Muscle.loss == "control"]
```

```
## class: SummarizedExperiment
## dim: 63 30
## metadata(0):
## assays(1): counts
## rownames(63): X1.6.Anhydro.beta.D.glucose X1.Methylnicotinamide ...
##   pi.Methylhistidine tau.Methylhistidine
## rowData names(0):
## colnames: NULL
## colData names(2): Patient.ID Muscle.loss
```

Como se indicaba en la descripción de los datos, hay 47 muestras de pacientes con caquexia y 30 muestras control.

Para seguir con la descripción estadística de los datos, calcularé la media, mediana, desviación estándar y rango de los metabolitos para ver la dispersión gneral de los datos.

```
# Cargar los datos de metabolitos
metabolite_data <- assays(se)$counts

# Convertir a data frame para facilitar el cálculo de estadísticas
metabolite_data_df <- as.data.frame(metabolite_data)

# Calcular estadísticas descriptivas
stats_summary <- data.frame(
  Metabolite = rownames(metabolite_data_df),
  Mean = apply(metabolite_data_df, 1, mean, na.rm = TRUE),
  Median = apply(metabolite_data_df, 1, median, na.rm = TRUE),
  SD = apply(metabolite_data_df, 1, sd, na.rm = TRUE),
  Range = apply(metabolite_data_df, 1, function(x) diff(range(x, na.rm = TRUE)))
)

# Mostrar las primeras filas del resumen estadístico
print(head(stats_summary))
```

```
##                                           Metabolite      Mean Median
## X1.6.Anhydro.beta.D.glucose X1.6.Anhydro.beta.D.glucose 105.63039  45.60
## X1.Methylnicotinamide             X1.Methylnicotinamide  71.57364  36.60
## X2.Aminobutyrate                       X2.Aminobutyrate  18.15974  10.49
## X2.Hydroxyisobutyrate             X2.Hydroxyisobutyrate  37.25065  32.46
## X2.Oxoglutarate                         X2.Oxoglutarate 145.08714  55.15
## X3.Aminoisobutyrate                 X3.Aminoisobutyrate  76.75636  22.65
##                                          SD   Range
## X1.6.Anhydro.beta.D.glucose 130.02560  680.69
## X1.Methylnicotinamide       133.19281 1026.35
```

```
## X2.Aminobutyrate             27.61453  171.15
## X2.Hydroxyisobutyrate        23.95681   88.84
## X2.Oxoglutarate             342.52217 2459.60
## X3.Aminoisobutyrate         191.01424 1477.69
```

Análisis de Componentes Principales

Como se ha explicando en la actividad 1.3, una de las mejores estrategias para explorar la variabilidad y patrones en los datos de metabolóica es realizando un PCA. Como se explica en la teoria, las primeras componentes nos permitiran enetender la mayor parte de la variabilidad ya que cada componente explica más que el siguiente.

Para eso, primero hay que preparar los datos, asegurarse que sean valores numéricos y que no haya valores NAs, transponer l matriz y escalar los datos.

```
library(magrittr)
```

```
##
## Adjuntando el paquete: 'magrittr'

## The following object is masked from 'package:GenomicRanges':
##
##     subtract
```

```
#Extraer los datos de metabolitos del objeto SummarizedExperiment
metabolite_data <- assays(se)$counts %>%
    as.matrix() %>%
    na.omit()

# Transponer la matriz de modo que las filas sean los pacientes y las columnas los metabolitos
metabolite_data_t <- t(metabolite_data)

#Escalar los datos
metabolite_data_scaled <- scale(metabolite_data_t)
```

Como se explica aquí https://aspteaching.github.io/AMVCasos/#ejemplo-2-an%C3%A1lisis-de-correspondencias-de-datos-de-microarrays, se puede hacer uso de la función `prcomp` para caclular las componentes principales.

```
# Calcular la PCA
pca_result <- prcomp(metabolite_data_scaled, center = TRUE, scale. = TRUE)

# Resumen de las 10 primeras PCs
pca_summary <- summary(pca_result)
pca_summary$importance[, 1:10]
```

```
##                            PC1      PC2      PC3      PC4      PC5      PC6
## Standard deviation     5.04667 2.270128 1.833107 1.747276 1.659056 1.613049
## Proportion of Variance 0.40427 0.081800 0.053340 0.048460 0.043690 0.041300
## Cumulative Proportion  0.40427 0.486070 0.539410 0.587870 0.631560 0.672860
##                            PC7      PC8      PC9     PC10
## Standard deviation     1.473038 1.364032 1.242747 1.206504
## Proportion of Variance 0.034440 0.029530 0.024510 0.023110
## Cumulative Proportion  0.707300 0.736830 0.761350 0.784450
```

```
pca_result$rotation[,1]
```

```
## X1.6.Anhydro.beta.D.glucose       X1.Methylnicotinamide
##                  0.07678198                  0.06448034
##            X2.Aminobutyrate       X2.Hydroxyisobutyrate
##                  0.11064656                  0.14196456
##             X2.Oxoglutarate         X3.Aminoisobutyrate
##                  0.08826605                  0.08984882
##          X3.Hydroxybutyrate       X3.Hydroxyisovalerate
##                  0.15920496                  0.13137204
##           X3.Indoxylsulfate     X4.Hydroxyphenylacetate
##                  0.11955834                  0.11156540
##                     Acetate                     Acetone
##                  0.10974322                  0.09241969
##                     Adipate                     Alanine
##                  0.10013298                  0.16734332
##                  Asparagine                     Betaine
##                  0.16916015                  0.12340673
##                   Carnitine                     Citrate
##                  0.08598625                  0.15729614
##                     Creatine                  Creatinine
##                  0.04147524                  0.17549735
##                Dimethylamine                Ethanolamine
##                  0.15905220                  0.17041813
##                     Formate                      Fucose
##                  0.13354188                  0.16297413
##                     Fumarate                     Glucose
##                  0.12617442                  0.06024052
##                    Glutamine                     Glycine
##                  0.17089565                  0.14621127
##                    Glycolate               Guanidoacetate
##                  0.10987898                  0.08813351
##                    Hippurate                    Histidine
##                  0.07137427                  0.15784801
##                 Hypoxanthine                  Isoleucine
##                  0.15450422                  0.13253073
##                     Lactate                     Leucine
##                  0.06170896                  0.16000426
##                      Lysine                  Methylamine
##                  0.07703294                  0.13408244
##               Methylguanidine         N.N.Dimethylglycine
##                  0.11636219                  0.13986242
##            O.Acetylcarnitine                 Pantothenate
##                  0.12430489                  0.05746210
##                Pyroglutamate                     Pyruvate
##                  0.15962554                  0.12796424
##                  Quinolinate                      Serine
##                  0.12137345                  0.16496675
##                    Succinate                     Sucrose
##                  0.14547514                  0.04240689
##                     Tartrate                     Taurine
##                  0.10752831                  0.10753145
##                    Threonine                Trigonelline
```

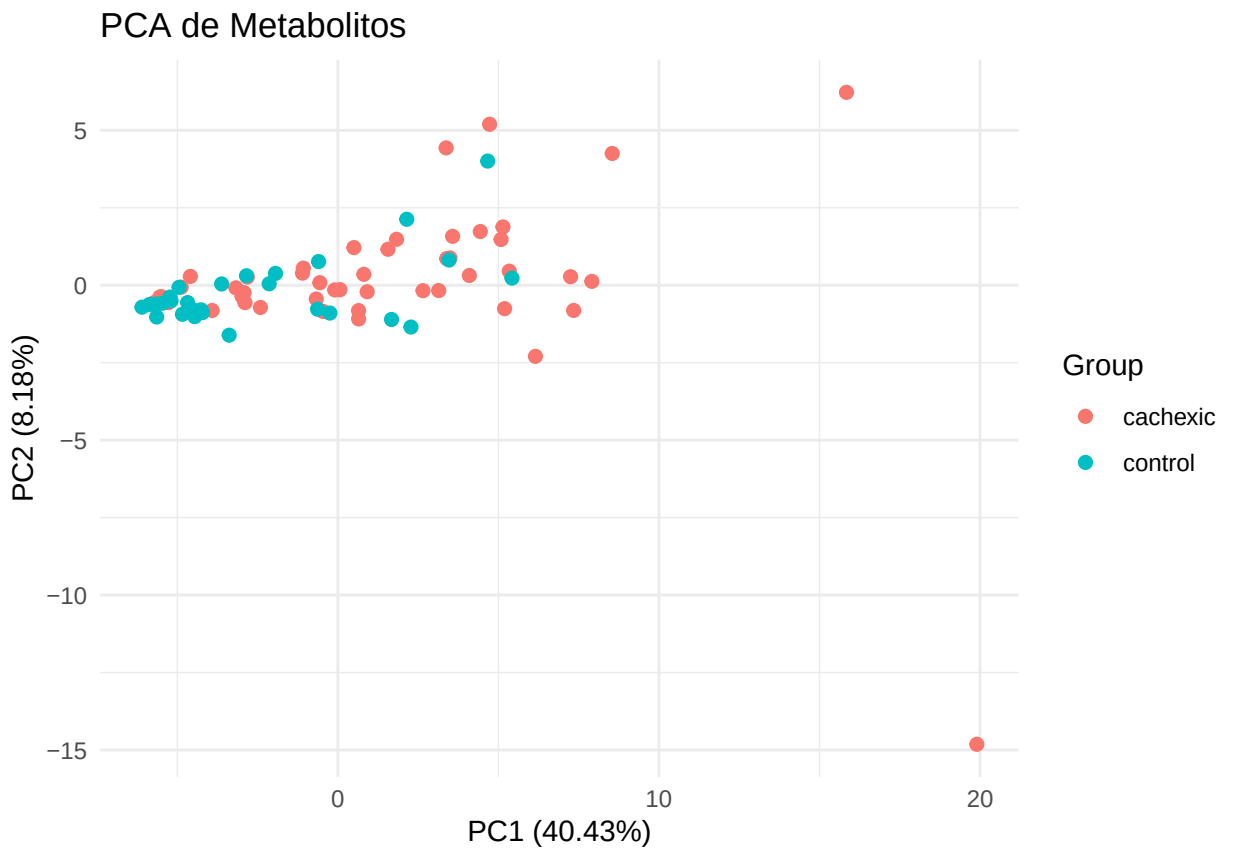
```
##                   0.16845973                   0.12528021
##      Trimethylamine.N.oxide                   Tryptophan
##                   0.08774914                   0.14292599
##                     Tyrosine                        Uracil
##                   0.16204954                   0.11849005
##                      Valine                       Xylose
##                   0.16791310                   0.04900948
##                cis.Aconitate                myo.Inositol
##                   0.16611699                   0.07609235
##              trans.Aconitate          pi.Methylhistidine
##                   0.12237835                   0.06520172
##         tau.Methylhistidine
##                   0.11957240
```

Con esta útima salida podemos describir la PC1:

$Y_i = 0.07678198$ x X1.6.Anhydro.beta.D.glucose + 0.06448034 x X1.Methylnicotinamide + 0.11064656 x X2.

Para visualizarlo en un gráfico, con `ggplot2` se pueden mostrar los dos primeros componentes principales, que, como se ha dicho anteriormente, suelen capturar la mayor parte de la variabilidad en los datos.

```r
library(ggplot2)

# Calcular el pocentaje de variabilidad explicada por cada componente
explained_variance <- pca_result$sdev^2 / sum(pca_result$sdev^2) * 100


# Convertir los datos de PCA a un data frame para graficar
pca_data <- as.data.frame(pca_result$x)

# Agregar la información de cachexia
pca_data$Group <- colData(se)$Muscle.loss

# Crear el gráfico
ggplot(pca_data, aes(x = PC1, y = PC2, color = Group)) +
    geom_point(size = 2) +
    labs(title = "PCA de Metabolitos",
         x = paste0("PC1 (", round(explained_variance[1], 2), "%)"),
         y = paste0("PC2 (", round(explained_variance[2], 2), "%)")) +
    theme_minimal()
```



El primer componte explica un 40.43% de las variabilidad. Se puede ver asociada al factor `Muscle.loss`, los pacientes con caquexia en general a la derecha y los control a la izquierda. Mientras que la segunda componte explica muy poca variabilidad, solo un 8.18%.

Boxplot de metabolitos por grupo

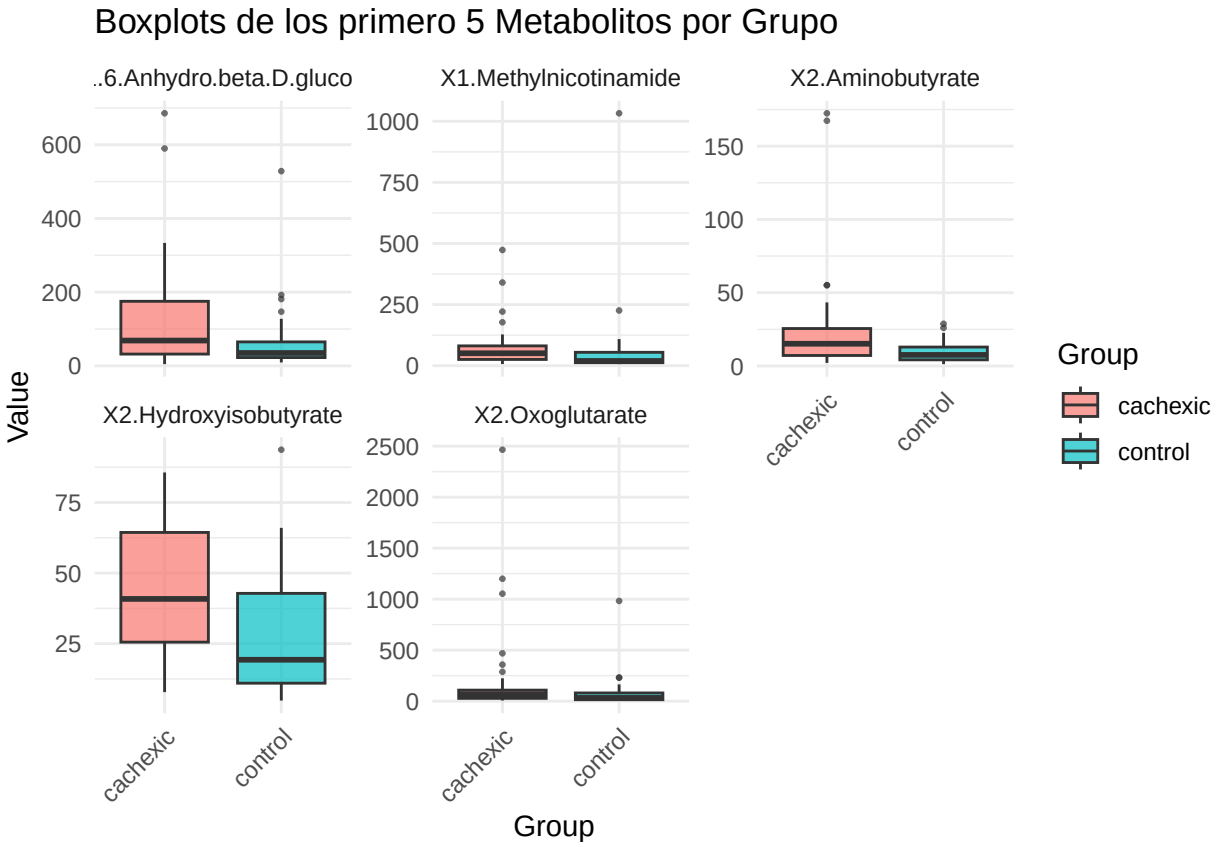
Otra manera interesante de explorar los datos en este tipo de experimentos donde hay dos grupos (caquexia y control) es realizr un boxplot por grupo para cada metabolito. De esta manera se puede ver como varia cada metabólito entre los grupos y por lo tanto ver que metabolitos sufren una modificación en sus valores cuando se este síndrome esta presente.

Si estuvieramos buscando un metabolito concreto nos centrariamos en ese, pero para poner un ejemplo, he representados los primeros 5 metabolitos que aparecen en la tabla.

```
library(tidyr)
```

```
##
## Adjuntando el paquete: 'tidyr'

## The following object is masked from 'package:magrittr':
##
##     extract

## The following object is masked from 'package:S4Vectors':
##
##     expand
```

```r
# Convertir los datos de metabolitos a formato largo
metabolite_data_long <- as.data.frame(metabolite_data_t)
metabolite_data_long$Group <- colData(se)$Muscle.loss  # Añadir grupo
metabolite_data_long <- pivot_longer(metabolite_data_long, cols = -Group, names_to = "Metabolite", valu

# Seleccionar los 5 primeros metabolitos
top5_metabolites <- head(unique(metabolite_data_long$Metabolite), 5)
metabolite_data_long <- metabolite_data_long[metabolite_data_long$Metabolite %in% top5_metabolites, ]

# Graficar boxplots
library(ggplot2)
ggplot(metabolite_data_long, aes(x = Group, y = Value, fill = Group)) +
  geom_boxplot(outlier.size = 0.5, alpha = 0.7) +
  facet_wrap(~ Metabolite, scales = "free_y") +
  labs(title = "Boxplots de los primero 5 Metabolitos por Grupo") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Observando este gráfico, interprtaríamos los resultados y los compararíamos con la bibliografía consultada. A simple vista, parece que hay un aumento en los valores de estos metabolitos en las muestras de los pacientes que presentan caquexia en comparación con los pacientes control, aunque desconocemos si esta diferencia es estadísticamente signficativa.

Discusión y limitaciones

Los resultados preliminarares obtenidos muestra que hay una variación en los perfiles de metabolitos entre los pacientes cachexicos y los controles, lo que sugiere que algunos metabolitos pueden actuar como biomarcadores potenciales. Sin embargo, este estudio tiene limitaciones, incluyendo el tamaño de la muestra y la posibilidad de variabilidad en las mediciones de metabolitos. Además, no se considera el efecto de factores confusos como la edad, el sexo o el tratamiento en curso de los pacientes.

Las conclusiones sugieren que el análisis de metabolitos puede proporcionar información valiosa para entender la fisiología de la cachexia, sin embargo, se recomienda que futuros estudios incluyan una mayor diversidad de pacientes y un enfoque longitudinal.

Repositorio github.

https://github.com/SofiaPetrissans/Petrissans-Moll-Sofia-PEC1

Contiene:

- el informe (Petrissans_Moll_Sofia_Informe.pdf)
- el objeto contenedor con los datos y los metadatos en formato binario (se_cachexia.Rda)
- el código R para la exploración de los datos (PEC1.Rmd)
- los datos en formato csv (human_cachezia.csv)
- los metadatos acerca del dataset en un archivo markdown (metadatos.Rmd)