Análisis de datos ómicos. Primera prueba de evaluación continua

Víctor Fructuoso Sánchez

Índice

Resumen **1**

Objetivos **1**

Métodos **2**

Resultados **2**

Discusión **9**

Conclusiones **10**

Referencias **10**

Anexo **10**

Resumen

El siguente estudio se centra en la metabolómica urinaria y su potencial para la detección del cáncer gástrico. Se lleva a cabo un proceso de análisis de datos ómicos simple, donde a partir de un conjunto de datos que contiene la concentración de una serie de metabolitos presentes en muestras de orina de distitnos tipos de pacientes, se realiza un análisis exploratorio de estos datos. Para ello, se crea un objeto del tipo SummarizedExperiment que contiene la información a analizar. Una vez se tiene el SummarizedExperiment creado, se procede a realizar un análisis exploratorio inicial para comprobar distintas características tales como el número de metabolitos, el número de muestras, la distribución de los datos. Por último, se realizan las transformaciones apropiadas para realizar un Análisis de Componentes Principales que permita comprobar la agrupación de las muestras estudiadas, resultando en una clara diferenciación del grupo control con respecto a las 3 clases de pacientes, además de una ligera diferenciación del grupo de pacientes con Cáncer Gástrico con respecto a los otros dos grupos de pacientes, que presentan un mayor grado de solapamiento, por último, se determinan cuáles son los metabolitos más influyentes en esta agrupación.

Objetivos

El objetivo principal de este proyecto es la exploración de un conjunto de datos metabolitos a través de un objeto tipo SummarizedExperiment y su comparación con el ExpressionSet para comprobar cómo se agrupan las muestras en función de la clase a la que pertenecen, control, individuos sanos, enfermedad gástrica benigna y cáncer gástrico.

Métodos

Se escoge el dataset "2023-CIMCBTutorial" que contiene un conjunto de datos con información sobre la concentración de metabolitos en muestras de orina en 3 clases distitnas, que corresponden con distintos tipos de pacientes: individuos sanos (HE), enfermedad gástrica benigna (BN) y cáncer gástrico (GC). Los datos se han obtenido de *[este repositorio de GitHub]*(https://github.com/nutrimetabolomics/metaboData), aunque también pueden encontrarse en la página Metabolomics Workbench bajo la ID PR000699. Para el análisis se utiliza el programa R, en concreto el paquete SummarizedExperiment y el paquete POMA, ambos de Bioconductor. El análisis se ha enfocado de la siguiente forma:

-Importación y primer vistazo a los datos, estructura y dimensiones.

-Preprocesado. Consiste en la eliminación e imputación de valores faltantes y normalización de los datos con el método "log_pareto"

-Análisis de Componentes Principales. Método de visualización de la separación de los grupos en base a la concentración de metabolitos.

Resultados

En primer lugar, se importan los datos y se cargan los paquetes adecuados para su análisis.

```
library(readxl)
# Importación de los datos
GastricCancer_NMR <- read_excel("GastricCancer_NMR.xlsx")
GastricCancer_NMR_Peak <- read_excel("GastricCancer_NMR.xlsx",
    sheet = "Peak")
# Se instala el paquete SummarizedExperiment a través de Bioconductor con
# BiocManager::install("SummarizedExperiment")
library(SummarizedExperiment)
```

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

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

```
##
## 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
```

```
## 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
```

```
library(POMA)
```

```
## Welcome to POMA!
## Version 1.12.0
## POMAShiny app: https://github.com/pcastellanoescuder/POMAShiny
## For more detailed package information please visit https://pcastellanoescuder.github.io/POMA/
```

Una vez cargados los datos, se pasa a la creación de un objeto de clase SummarizedExperiment. Este tipo de objeto es una clase del paquete Bioconductor que permite almacenar matrices de datos junto con metadatos adicionales que describen la matriz de datos. Por tanto, un objeto de este tipo contiene:

1. Una o más matrices de datos (assay) que contiene los valores de expresión, en este caso serán las concentraciones de metabolitos en las muestras. Equivalente a exprs() en un ExpressionSet
2. Un conjunto de metadatos, que contiene infromación sobre las muestras. Equivalente a pData en un ExpressionSet
3. Otro conjunto de metadatos, que contiene información sobre las filas de la matriz de datos, equivalente a fData en un ExpressionSet
4. Información adicional opcional.

Se observa que es un tipo de objeto muy similar a un ExpressionSet, con la diferencia principal de que el SummarizedExperiment es capaz de almacenar varias matrices de datos.

```
#Creación del objeto SummarizedExperiment a partir de los datos importados
#Incorporación del "rowData"
row_Data<-data.frame(GastricCancer_NMR_Peak)
#Incorporación del assay
matriz_datos<-as.matrix(GastricCancer_NMR[, 5:153])
#Incorporación de colData, usando como nombres de fila el SampleID
metadatos<-data.frame(GastricCancer_NMR[, 1:4], row.names = GastricCancer_NMR$SampleID)
#Es necesario trasponer la matriz para una correcta lectura de los datos de metabolitos
```

```
matrizT<-t(matriz_datos)
#Creación del SummarizedExperiment(SE)
SE_GastricCancerNMR<-SummarizedExperiment(
  assays = list(metabolitos = matrizT),
  colData = metadatos,
  rowData = row_Data
)
SE_GastricCancerNMR
```

```
## class: SummarizedExperiment
## dim: 149 140
## metadata(0):
## assays(1): metabolitos
## rownames(149): M1 M2 ... M148 M149
## rowData names(5): Idx Name Label Perc_missing QC_RSD
## colnames(140): sample_1 sample_2 ... sample_139 sample_140
## colData names(4): Idx SampleID SampleType Class
```

Ahora se procede a obtener información de este objeto para conocer las características del conjunto de datos En el objeto SummarizedExperiment creado se observa que contiene 149 filas, que se corresponden con cada metabolito, de las cuales se dispone de información adicional en rowData. Por otro lado, el objeto presenta 140 columnas, donde cada una se corresponde con una muestra del conjunto de datos. Por lo tanto, el conjunto de datos contiene 140 muestras, donde de cada muestra se mide la concentración de 149 metabolitos.

Los componentes del SummarizedExperiment se pueden consultar por separado en el Anexo.

Con el comando rowData se observa la información de los metabolitos. En este caso, disponemos de una ID, que se corresponde con cada metabolito, el nombre del metabolito, siguiendo el mismo criterio que en el assay "metabolitos", una etiqueta con el metabolito, y unos valores de Perc_missing, que indican el porcentaje de valores faltantes por cada metabolito y QC_RSD, que muestra un coeficiente de variación con respecto a las muestras control (QC)

Al extraer el colData, se obtiene información de cada muestra, donde se encuentra una ID para cada muestra, el tipo de muestra (si es o no una muestra control) y la clase.

También se puede extraer la información de la concentración de metabolitos en cada una de las muestras. En este caso se observan las primeras líneas y columnas del conjunto de datos. A continuación se explora si el conjunto de datos contiene valores faltantes y cuales son los metabolitos con mayor porcentaje de NA, esto puede comprobarse ordenando los valores en función de la columna Perc_missing, presente en rowData, lo que nos permite saber cuales son los metabolitos con menor número de muestras.

```
#Comprobación de valores faltantes
sum(is.na(assays(SE_GastricCancerNMR)$metabolitos))
```

```
## [1] 1069
```

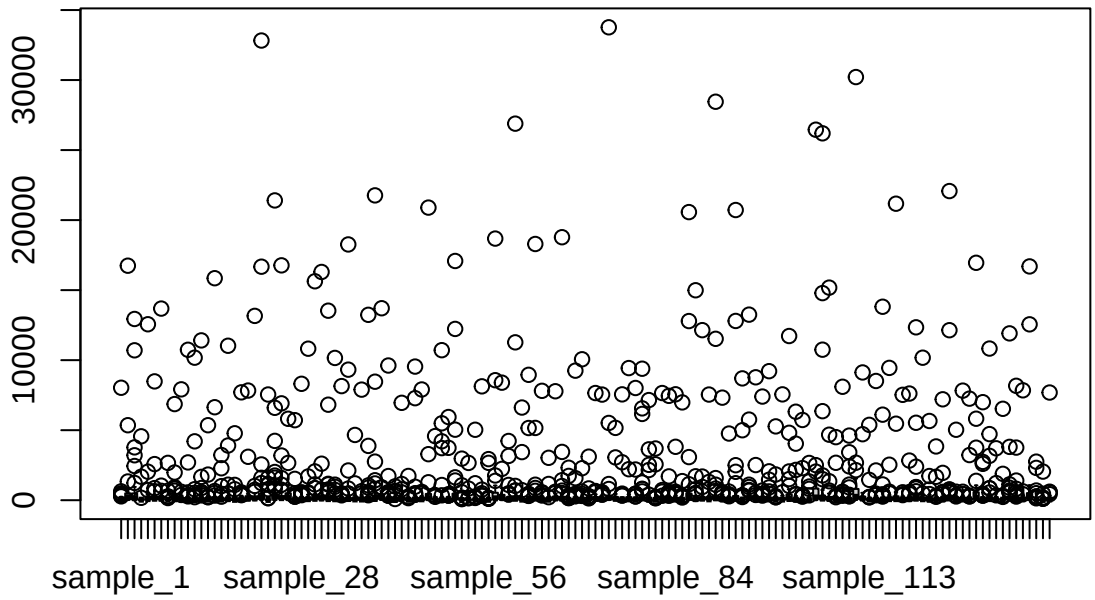
```
#Metabolitos con más valores faltantes.
rowData(SE_GastricCancerNMR)[order(-rowData(SE_GastricCancerNMR)$Perc_missing),]
```

```
## DataFrame with 149 rows and 5 columns
##            Idx        Name                  Label Perc_missing    QC_RSD
##      <numeric> <character>            <character>    <numeric> <numeric>
## M21         21         M21 4-Hydroxyphenylacetate      31.4286   65.4431
```

```
## M79          79         M79                  Lysine      29.2857   45.2577
## M17          17         M17 3-Hydroxyphenylacetate      27.1429  133.7857
## M95          95         M95 N-Phenylacetylphenyl..      25.7143   74.2231
## M145        145        M145                   uarm1      23.5714   41.4070
## ...         ...         ...                     ...         ...       ...
## M132        132        M132            u122triplet           0  29.18475
## M133        133        M133                    u14           0  21.05291
## M135        135        M135             u14doublet           0  26.42398
## M144        144        M144                    u87           0   6.63549
## M149        149        M149      -Methylhistidine            0   8.35180
```

Se observa que existen un total de 1069 valores faltantes, donde el metabolito M21 (4-Hydroxyphenylacetate) es el que presenta un mayor porcentaje (31.43% de valores faltantes, aproximadamente) Se muestra a continuación un boxplot para comprobar la distribución de los datos

```
boxplot(assays(SE_GastricCancerNMR)$metabolitos[1:50,])
```



Para una primera representación, se han dividido los metabolitos en grupos (del 1 al 50, del 50 al 100 y del 100 al 149, los dos boxplot restantes pueden consultarse en el Anexo). Se observa una elevada variabilidad entre la concentración de metabolitos, donde la mayoría de datos parecen encontrarse muy cercanos al 0, por lo que se procede a realizar un preprocesamiento de los datos mediante el paquete POMA, también de Bioconductor. El preprocesado que se realiza en los datos consiste en una imputación de los valores NA, usando un método knn, y una normalización de los datos, con un umbral de eliminación de los valores faltantes del 20%.

```
#Imputación de valores NA en el SummarizedExperiment mediante POMA
imputed_SE_GastricCancerNMR<-PomaImpute(SE_GastricCancerNMR, ZerosAsNA = TRUE, RemoveNA = TRUE,
                                        cutoff= 20, method ="knn")
#Normalización de los datos imputados utilizando el método log_pareto
norm_SE_GastricCancerNMR<-PomaNorm(imputed_SE_GastricCancerNMR, method ="log_pareto")
norm_SE_GastricCancerNMR
```

```
## class: SummarizedExperiment
## dim: 149 140
## metadata(0):
## assays(1): ''
## rownames(149): M1 M2 ... M148 M149
## rowData names(0):
## colnames(140): sample_1 sample_2 ... sample_139 sample_140
## colData names(4): Idx SampleID SampleType Class
```

```
# Se comprueba que no existen valores faltantes tras el preprocesado
sum(is.na(norm_SE_GastricCancerNMR@assays@data@listData[[1]]))
```

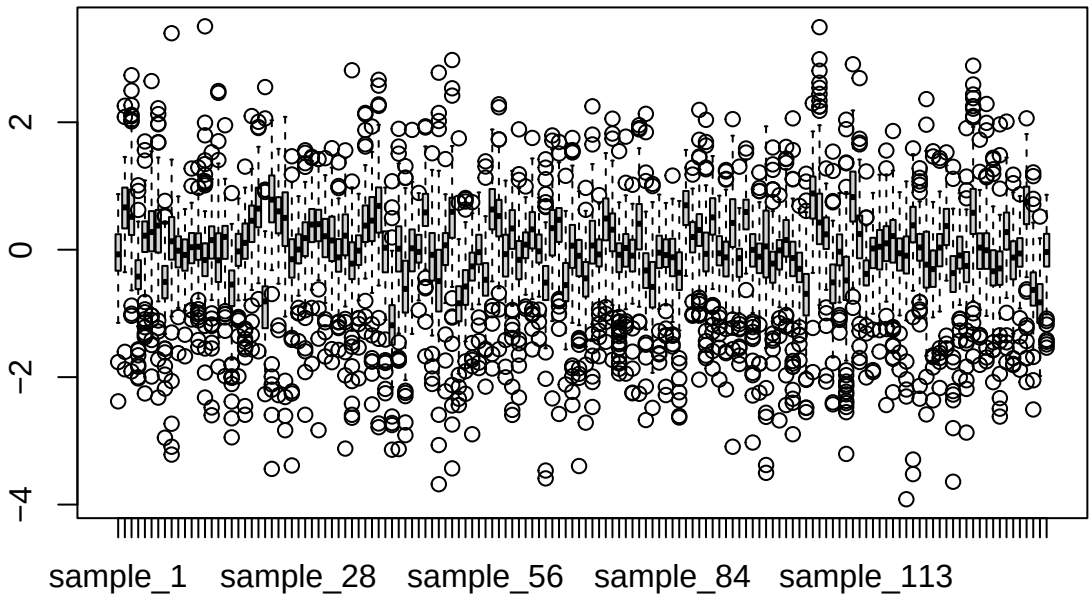
```
## [1] 0
```

Se observa que al tratar el SummarizedExperiment con las funciones de POMA, pierden la información asignada a rowData. Dado que las dimensiones del SummarizedExperiment se mantienen, es posible añadir de nuevo la información de los metadatos. El nombre del assay también se ha perdido, pero se puede guardar la información en una nueva variable. A continuación, se realiza de nuevo el boxplot con los datos normalizados

```
rowData(norm_SE_GastricCancerNMR)<-row_Data
norm_SE_GastricCancerNMR
```

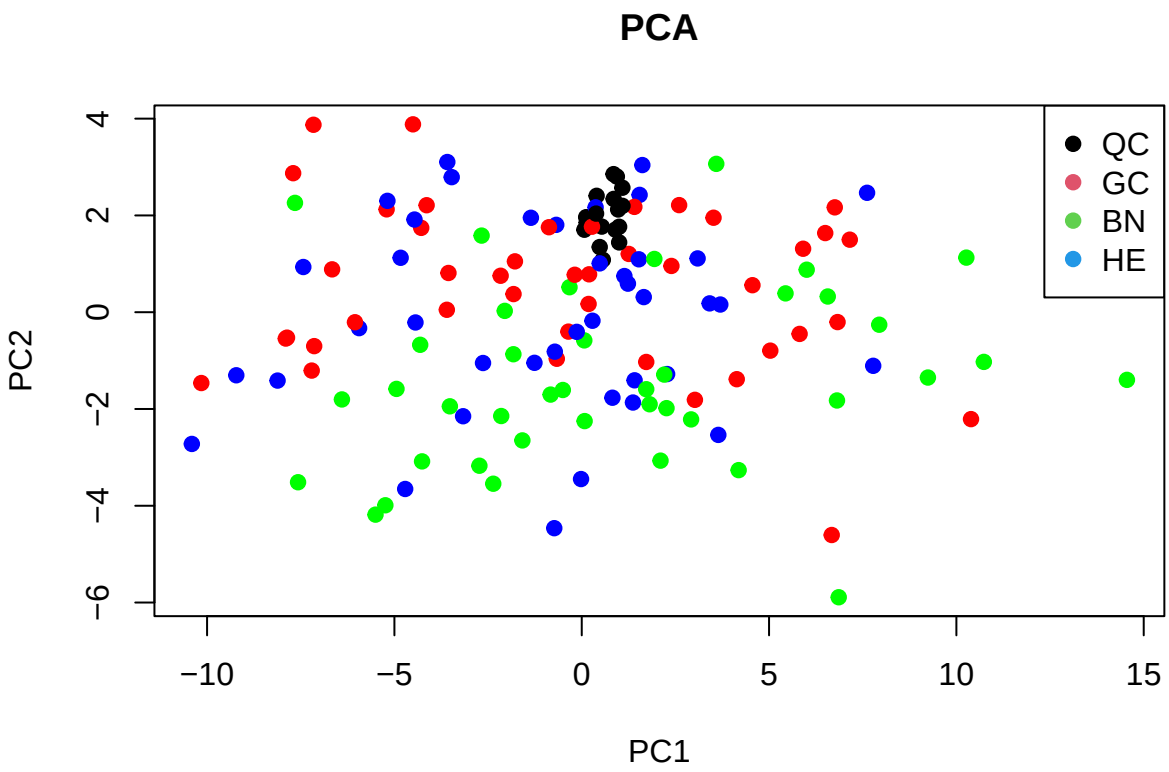
```
## class: SummarizedExperiment
## dim: 149 140
## metadata(0):
## assays(1): ''
## rownames(149): M1 M2 ... M148 M149
## rowData names(5): Idx Name Label Perc_missing QC_RSD
## colnames(140): sample_1 sample_2 ... sample_139 sample_140
## colData names(4): Idx SampleID SampleType Class
```

```
matriz_concentraciones_norm<-norm_SE_GastricCancerNMR@assays@data@listData[[1]]
boxplot(matriz_concentraciones_norm)
```



El nuevo boxplot presenta una distribución mucho más simétrica que el anterior, por lo que la transformación de los datos ha sido adecuada, y aunque todavía se observan algunos outliers, se han reducido en comparación con el boxplot anterior.

```
#Creación del PCA
pca<-prcomp(t(matriz_concentraciones_norm), scale = FALSE)
#Creación de una paleta de colores para una mejor visualización de los puntos
paleta<-c("GC" = "red", "BN" = "blue", "HE" = "green", "QC" = "black")
#Representación del PCA
plot(pca$x[,1], pca$x[,2],
     col = paleta[colData(norm_SE_GastricCancerNMR)$Class],
     xlab = "PC1", ylab = "PC2", pch = 19, main = "PCA")
legend("topright", legend = unique(colData(norm_SE_GastricCancerNMR)$Class),
       col = 1:length(unique(colData(norm_SE_GastricCancerNMR)$Class)), pch = 19)
```



En un primer vistazo, se observa que el grupo control (QC) se encuentra agrupado en el centro del gráfico, lo que indica que estas muestras son muy similares entre sí, por lo que es un resultado esperable. El resto de grupos se encuentran más o menos superpuestos entre sí, mostrando el grupo de cáncer gástrico (GC) una ligera separación de los otros dos. El siguiente paso en el análisis consiste en comprobar cuales son los metabolitos que tienen un mayor peso en el PCA.

```
m_pca<-pca$rotation
metabolitos_pc1 <- rownames(m_pca)[order(abs(m_pca[, 1]), decreasing = TRUE)]
head(metabolitos_pc1)
```

```
## [1] "M65"  "M104" "M53"  "M60"  "M5"   "M108"
```

```
metabolitos_pc2 <- rownames(m_pca)[order(abs(m_pca[, 2]), decreasing = TRUE)]
head(metabolitos_pc2)
```

```
## [1] "M136" "M139" "M145" "M45"  "M13"  "M24"
```

Se observa que los metabolitos más influyentes son, por un lado M65, M104, M53, M60, M5 y M108 para uno de los componentes, y para otro componente son M136, M139, M145, M45, M13 y M24.

Discusión

En el estudio se ha utilizado un objeto tipo SummarizedExperiment, que permite almacenar y gestion los datos de metabolitos, lo cual ha facilitado la integración de los datos de estudio, que se corresponden con

concentraciones de metabolitos, con los metadatos, que aportan una información de interés tanto para las muestras como para los metabolitos, lo cual es de gran utilidad a la hora de interpretar los resultados. Una vez realizado el análisis, se observa una clara separación del grupo control, lo que confirma la calidad de los datos dado que es un resultado esperable. En cuanto al resto de grupos, se observa una ligera separación del grupo GC (Cáncer Gástrico) con respecto a los otros dos grupos, enfermedad benigna (BN) y pacientes sanos (HE), los cuales muestran un mayor grado de solapamiento. En base a este resultado, podría concluirse que la metabolómica urinaria podría ser un factor de interés en cuanto a la detección del cáncer gástrico. Por otro lado, al haber registrado los metabolitos más influyentes en la diferenciación de estos grupos al realizar el PCA, podría dar lugar a una línea de investigación para determinar biomarcadores relevantes en la detección del Cáncer Gástrico.

Conclusiones

Los objetos tipo SummarizedExperiment, junto con paquetes que facilitan su manejo como POMA, con componentes de Bioconductor que suponen un gran avance para el manejo e interpretación de los datos ómicos. En este caso, se han utilizado estos objetos para la elaboración de un análisis de componentes principales que permite la agrupación de muestras de distintos grupos de pacientes, pacientes con cáncer gástrico, enfermedad benigna, sanos y un grupo control. El PCA ha mostrado que el grupo control se encuentra claramente diferenciado, el grupo de cáncer gástrico se muestra ligeramente diferenciado con respecto a los individuos sanos y con enfermedad benigna, y estos dos últimos grupos muestran un mayor grado de solapamiento. Se determinan también los metabolitos que tienen una mayor importancia en el PCA, dando lugar a posibles biomarcadores que permitan la detección del cáncer gástrico.

Referencias

Chan, A. W., Mercier, P., Schiller, D., Bailey, R., Robbins, S., Eurich, D. T., Sawyer, M. B., Broadhurst, D. (2016). 1H-NMR urinary metabolomic profiling for diagnosis of gastric cancer. British Journal of Cancer, 114(1), 59-62. doi:10.1038/bjc.2015.414

Bioconductor. (2023). POMA: Preprocessing and statistical analysis of metabolomics data. Recuperado de https://www.bioconductor.org/packages/release/bioc/vignettes/POMA/inst/doc/POMA-workflow.html

Bioconductor. (2023). SummarizedExperiment for Coordinating Experimental Assays, Samples, and Regions of Interest. Recuperado de https://www.bioconductor.org/packages/release/bioc/vignettes/SummarizedExperiment/inst/doc/SummarizedExperiment.html

Enlace al repositorio en GitHub: https://github.com/vfructuoso/Fructuoso-Sanchez-Victor-PEC1.git

Anexo

```
dim(SE_GastricCancerNMR) #Dimensiones del SummarizedExperiment
```

```
## [1] 149 140
```

```
rowData(SE_GastricCancerNMR)
```

```
## DataFrame with 149 rows and 5 columns
##            Idx        Name                  Label Perc_missing    QC_RSD
```

```
##      <numeric> <character>           <character>    <numeric> <numeric>
## M1           1          M1     1_3-Dimethylurate    11.428571  32.20800
## M2           2          M2 1_6-Anhydro- -D-gluc..     0.714286  31.17803
## M3           3          M3   1_7-Dimethylxanthine     5.000000  34.99060
## M4           4          M4   1-Methylnicotinamide     8.571429  12.80420
## M5           5          M5         2-Aminoadipate     1.428571   9.37266
## ...        ...         ...                    ...          ...       ...
## M145       145        M145                  uarm1     23.57143   41.4070
## M146       146        M146                  uarm2      4.28571   34.4582
## M147       147        M147              -Alanine      1.42857   27.6235
## M148       148        M148      -Methylhistidine      1.42857   16.5619
## M149       149        M149      -Methylhistidine      0.00000    8.3518
```

```
colData(SE_GastricCancerNMR)
```

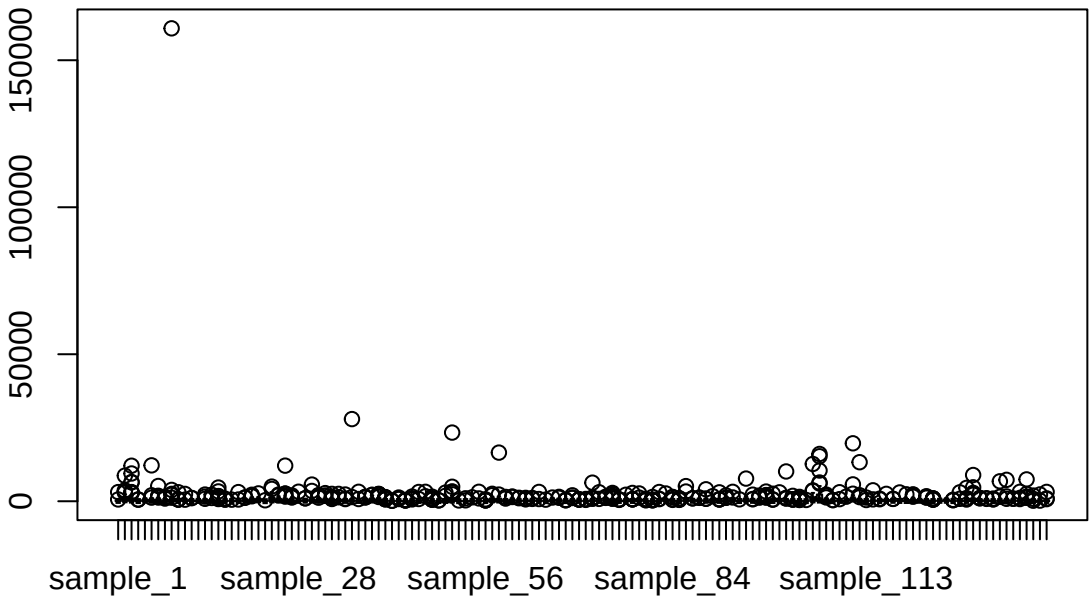
```
## DataFrame with 140 rows and 4 columns
##                   Idx    SampleID  SampleType       Class
##             <numeric> <character> <character> <character>
## sample_1            1    sample_1          QC          QC
## sample_2            2    sample_2      Sample          GC
## sample_3            3    sample_3      Sample          BN
## sample_4            4    sample_4      Sample          HE
## sample_5            5    sample_5      Sample          GC
## ...               ...         ...         ...         ...
## sample_136        136  sample_136          QC          QC
## sample_137        137  sample_137      Sample          GC
## sample_138        138  sample_138      Sample          BN
## sample_139        139  sample_139      Sample          HE
## sample_140        140  sample_140          QC          QC
```

```
head(assays(SE_GastricCancerNMR)$metabolitos)[,1:8] #Primeras filas de la matriz de datos
```

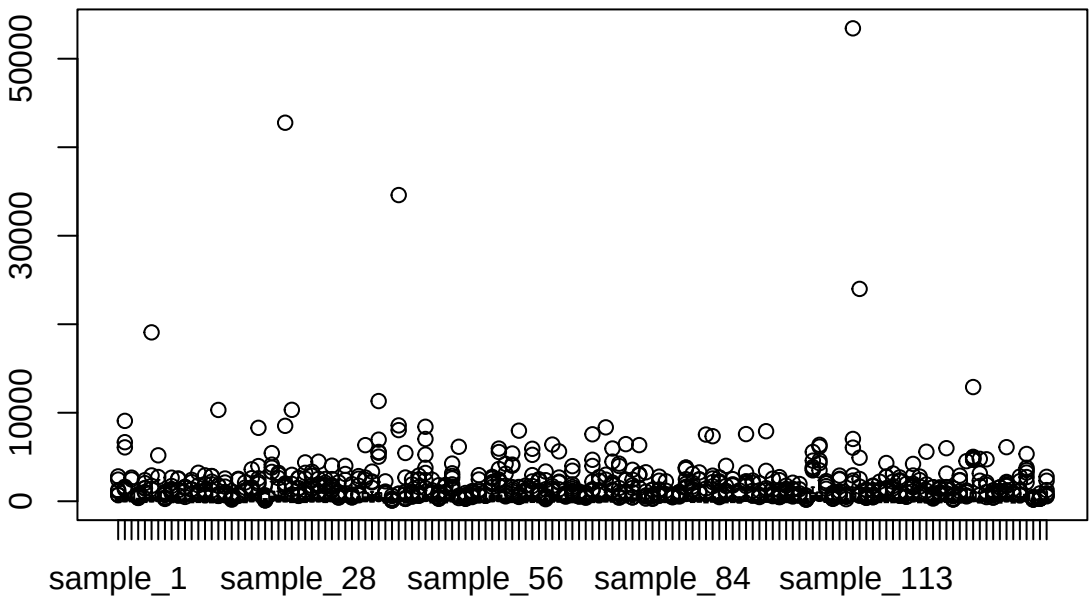
```
##    sample_1 sample_2 sample_3 sample_4 sample_5 sample_6 sample_7 sample_8
## M1     90.1     43.0    214.3     31.6     81.9    196.9     45.5     91.0
## M2    491.6    525.7  10703.2     59.7    258.7    128.2    190.4    231.9
## M3    202.9    130.2    104.7     86.4    315.1    862.5     32.0    212.5
## M4     35.0       NA     46.8     14.0      8.7     18.7       NA     18.2
## M5    164.2    694.5    483.4     88.6    243.2    200.1    362.7     72.5
## M6     19.7    114.5    152.3     10.3     18.4      4.7     35.7      6.7
```

```
#Se extraen únicamente las 8 primeras columnas para facilitar su lectura
```

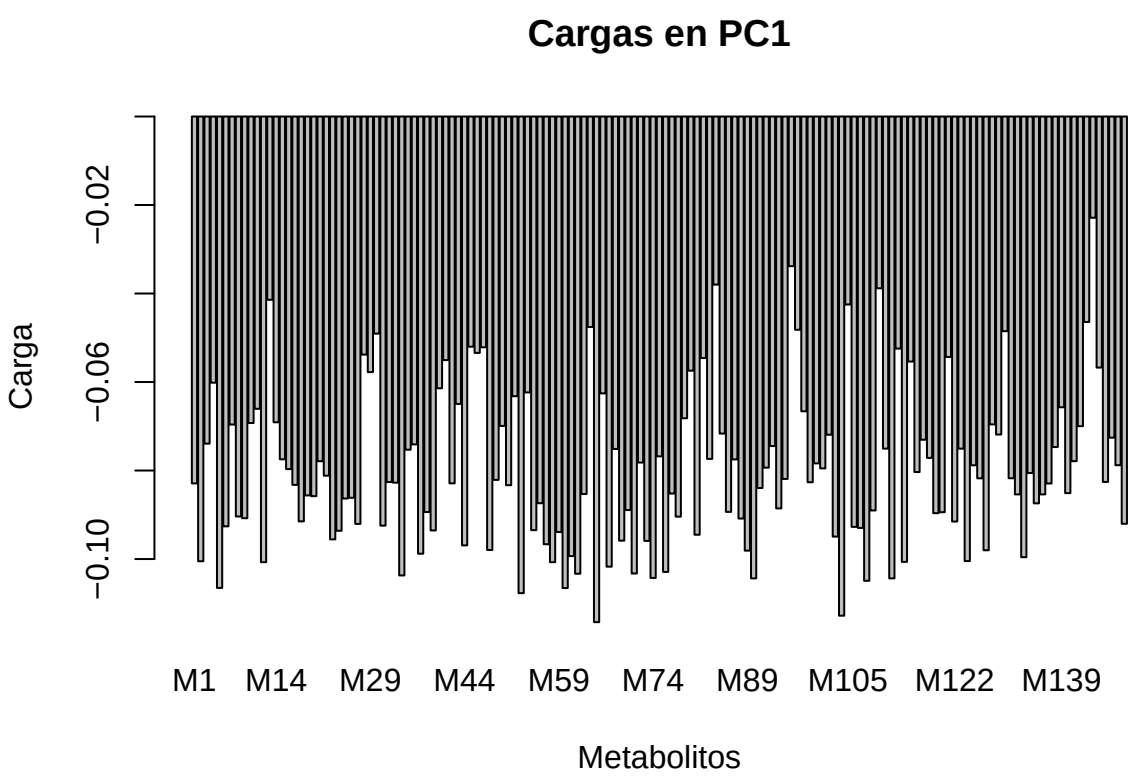
```
#Boxplot extra, para completar todos los metabolitos
boxplot(assays(SE_GastricCancerNMR)$metabolitos[51:100,])
```



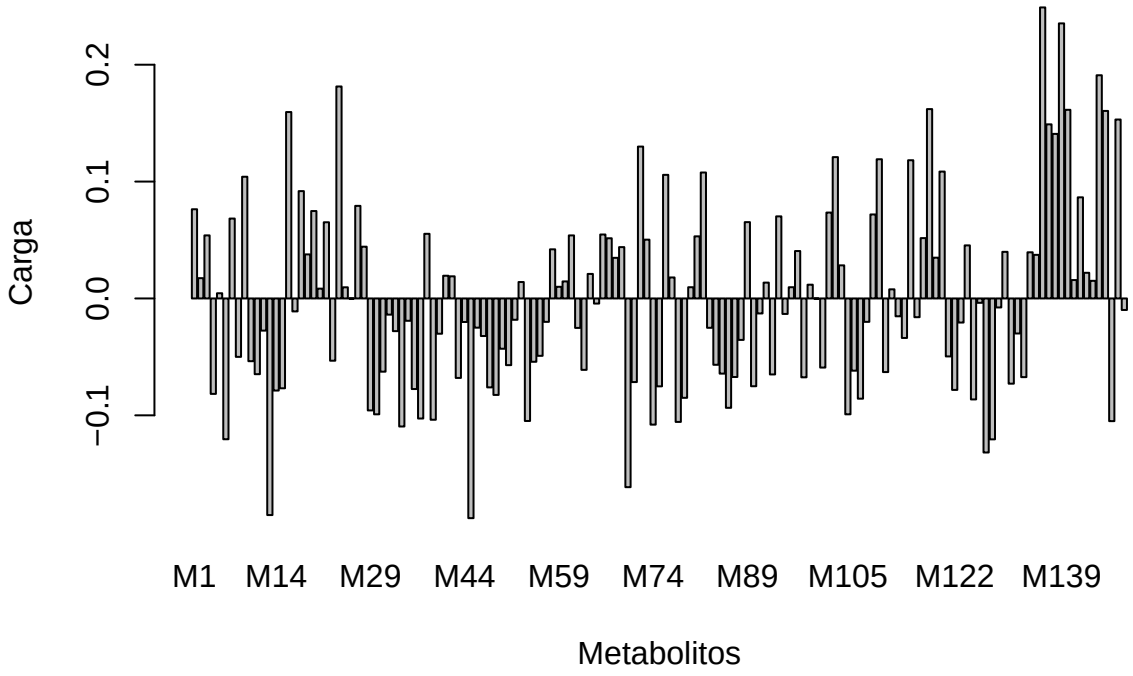

```
boxplot(assays(SE_GastricCancerNMR)$metabolitos[100:149,])
```




```
#Peso de los metabolitos en cada componente
barplot(m_pca[, 1], main = "Cargas en PC1", ylab = "Carga", xlab = "Metabolitos")
```




```r
barplot(m_pca[, 2], main = "Cargas en PC2", ylab = "Carga", xlab = "Metabolitos")
```


Cargas en PC2
Carga
0.2
0.1
0.0
−0.1
M1
M14
M29
M44
M59
M74
M89
M105
M122
M139
Metabolitos