PEC1 ADO

Beatriz Nistal Nuño

Tabla de contenidos

1. Abstract

Antecedentes: La metabolómica ha demostrado ser prometedora en la detección del cáncer gástrico (GC). Los datos para este análisis de datos ómicos se obtuvieron de un estudio de investigación que buscaba identificar si el GC presenta un perfil metabolómico urinario único en comparación con la enfermedad gástrica benigna (BN) y los pacientes sanos (HE) [1].

Métodos: el dataset de metabolómica seleccionado, con Project ID: PR000699, corresponde al proyecto *1H-NMR urinary metabolomic profiling for diagnosis of gastric cancer* [1]. Los datos importados constan de 140 muestras de orina y 149 metabolitos analizados en cada muestra. Las muestras de orina pertenecían a pacientes con GC, BN y HE. Se emplearon estadísticas univariantes y multivariantes para la evaluación de calidad de los datos. Previamente al análisis exploratorio se realizó una limpieza de datos. Las herramientas bioinformáticas utilizadas para la exploración de datos fueron R (R version 4.2.1 (2022-06-23 ucrt), Running under: Windows 10 x64 (build 19045)), Bioconductor (Biobase_2.58.0, BiocGenerics_0.44.0, BiocManager_1.30.25), y RMarkdown (rmarkdown_2.14)).

Resultados: Se pudo apreciar en el gráfico boxplot con todas las muestras que las muestras de GC parecen tener valores de concentraciones más extremos que las demás muestras. Todas las visualizaciones coincidieron en mostrar que los datos son de alta calidad ya que las muestras de controles de calidad (QC) se agrupan estrechamente en comparación con las muestras biológicas.

Conclusiones: el análisis exploratorio de los datos mostró que los datos son de alta calidad.

2. Objetivos

Llevar a cabo un análisis exploratorio que proporcione una visión general del dataset y realizar una evaluación de calidad del archivo de datos depositado en el repositorio de datos de Metabolomics Workbench con ID de proyecto PR000699 ("GastricCancer_NMR.xlsx") [1].

3. Métodos

He Seleccionado y descargado el dataset de metabolómica con Project ID: PR000699, que obtuve de metabolomicsWorkbench. El *Project Title* es: *1H-NMR urinary metabolomic profiling for diagnosis of gastric cancer* (Species: Homo sapiens). La justificación de la selección del dataset es que me ha interesado este estudio porque me parece una contribución interesante hacia el avance del diagnóstico del cáncer gástrico. La identificación de un perfil metabolómico urinario distintivo para el GC podría ofrecer una modalidad no invasiva, rentable, eficiente y razonablemente precisa para obtener diagnósticos precisos y poder discriminar entre pacientes con GC, sanos y con enfermedad gástrica benigna [1].

El tipo de los datos obtenidos de metabolomicsWorkbench es en formato excel, que he convertido a un objeto de clase SummarizedExperiment que contiene los datos y los metadatos.

El análisis exploratorio y evaluación de calidad de los datos los he realizado mediante un Análisis estadístico univariante, y una Visualización de datos multivariante que incluye Cálculo de las componentes principales (PCA), Técnicas de clustering con Agrupamiento jerárquico, y un Análisis basado en distancias. Para ello he incorporado los datos a un summarizedExperiment, y previamente al análisis he realizado una limpieza de datos. He creado un repositorio de GitHub que contiene toda la información del análisis llevado a cabo y los ficheros [2]. Las herramientas bioinformáticas que he utilizado para la exploración de datos son R (R version 4.2.1 (2022-06-23 ucrt), Platform: x86_64-w64-mingw32/x64 (64-bit), Running under: Windows 10 x64 (build 19045)), Bioconductor (Biobase_2.58.0, BiocGenerics_0.44.0, BiocManager_1.30.25), y RMarkdown (rmarkdown_2.14)).

4. Resultados

4.1. Creación de un objeto de clase SummarizedExperiment

Los datos se cargan como un archivo de Microsoft Excel, (cada columna es una variable y cada fila es una observación). El archivo de Excel contiene una Hoja de Datos y una Hoja de metabolitos. La Hoja de Datos contiene todas las concentraciones de metabolitos y los metadatos asociados a cada observación (incluye las columnas: Idx, SampleID, SampleType y Class). La Hoja de metabolitos contiene todos los metadatos correspondientes a cada metabolito medido (incluye las columnas: Idx, Name, Label, Perc_missing y QC_RSD).

En este ejemplo, los datos importados constan de 140 muestras y 149 metabolitos. Cada fila describe una sola muestra de orina, donde:

- Las columnas M1 a M149 describen las concentraciones de metabolitos.
- La columna *"SampleType"* indica si la muestra fue un control de calidad agrupado (QC) o una muestra de estudio.
- La columna *"Class"* indica el resultado clínico observado para ese individuo: GC = Cáncer gástrico, BN = Tumor benigno, HE = Control sano.

```
library(Biobase)
```

```
library(readxl)
datafile = read_excel("GastricCancer_NMR (final).xlsx", sheet=1) # leo la Hoja de Datos
datafile_matrix= as.matrix(datafile)  # convierto la hoja de excel a matriz
rownames(datafile_matrix)=datafile_matrix[ , 2]  #asigno a los nombres de las filas la columna del número de muestra
datafile_matrix_reduced=datafile_matrix[ ,c(-1,-2,-3,-4)]  # elimino las columnas que contienen los metadatos de cada muestra

# Transposing the matrix
t_datafile_matrix_reduced <- t(datafile_matrix_reduced)
dim(t_datafile_matrix_reduced)

## [1] 149 140

numbers=as.numeric(t_datafile_matrix_reduced[ ,1:140])  # convierto la matriz de concentraciones de metabolitos a números analizables
t_datafile_matrix_reduced= matrix(numbers, nrow=149, byrow=FALSE)

rownames(t_datafile_matrix_reduced)=colnames(datafile_matrix_reduced)  # asigno nombres a las filas
colnames(t_datafile_matrix_reduced)=rownames(datafile_matrix_reduced)  # asigno nombres a las columnas
t_datafile_matrix_reduced[1:5,1:5]

##    sample_1 sample_2 sample_3 sample_4 sample_5
## M1     90.1     43.0    214.3     31.6     81.9
## M2    491.6    525.7  10703.2     59.7    258.7
## M3    202.9    130.2    104.7     86.4    315.1
## M4     35.0       NA     46.8     14.0      8.7
## M5    164.2    694.5    483.4     88.6    243.2

sample_metadata = datafile_matrix[ ,c(1,2,3,4)]  # selecciono los metadatos de las muestras que luego formarán parte del objeto SummarizedExperiment
rownames(sample_metadata)=sample_metadata[ , 1] # asigno nombres a las filas
sample_metadata= sample_metadata[,-1]  # elimino la primera columna
sample_metadata[1:5,]

##     SampleID   SampleType Class
##   1 "sample_1" "QC"       "QC"
##   2 "sample_2" "Sample"   "GC"
##   3 "sample_3" "Sample"   "BN"
##   4 "sample_4" "Sample"   "HE"
##   5 "sample_5" "Sample"   "GC"

dim(sample_metadata)

## [1] 140   3
```

En este ejemplo, los datos importados de los metabolitos constan de 149 metabolitos (los mismos que en la tabla de datos).

Cada fila describe un único metabolito, donde:

- La columna Idx es un índice único de metabolito.
- La columna *Name* es el encabezado de columna correspondiente a este metabolito en la tabla de datos.
- La columna *Label* proporciona un nombre único para el metabolito (o un identificador).
- La columna *Perc_missing* indica el porcentaje de muestras que no contienen una medición de este metabolito (datos faltantes).
- La columna *QC_RSD* es una puntuación de calidad que representa la variación en las mediciones de este metabolito en todas las muestras.

```
metabolite_metadata <- read_excel("GastricCancer_NMR (final).xlsx", sheet=2 )  # leo la Hoja de metabolitos
metabolite_metadata=as.matrix(metabolite_metadata)  # convierto la hoja de excel a matriz

rownames(metabolite_metadata)=metabolite_metadata[,1]
metabolite_metadata=metabolite_metadata[,-1]  # elimino la primera columna

sample_metadata= as.data.frame(sample_metadata) # convierto la matriz de metadatos a dataframe, que es el tipo de datos que necesita el SummarizedExperiment
metabolite_metadata= as.data.frame(metabolite_metadata) # convierto la matriz de metabolitos a dataframe, que es el tipo de datos que necesita el SummarizedExperiment

sapply(metabolite_metadata, class)  # compruebo los tipos de datos de las columnas de los metabolitos

##         Name        Label Perc_missing       QC_RSD
##  "character"  "character"  "character"  "character"

metabolite_metadata$Perc_missing=as.numeric(metabolite_metadata$Perc_missing)  # convierto esta columna a numérica
metabolite_metadata$QC_RSD=as.numeric(metabolite_metadata$QC_RSD)  # convierto esta columna a numérica
```

Creo a continuación un objeto de clase SummarizedExperiment que contenga los datos y los metadatos (información acerca del dataset, sus filas y columnas). La clase SummarizedExperiment es una extensión de ExpressionSet.

La clase SummarizedExperiment es muy similar a ExpressionSet. La principal diferencia radica en que SummarizedExperiment está diseñada para almacenar mediciones asociadas con las ubicaciones del genoma. SummarizedExperiment utiliza un conjunto de argumentos diferente para almacenar esencialmente las mismas tablas. ExpressionSet se utiliza generalmente para *array-based experiments*, donde las filas son *features*, y SummarizedExperiment se utiliza generalmente para experimentos basados en secuenciación, donde las filas son GenomicRanges.

SummarizedExperiment es un contenedor de tipo matriz donde las filas representan *features* de interés (p. ej., genes, transcritos, exones, etc.) y las columnas representan muestras. Los objetos contienen uno o más *assay*, cada uno representado por un objeto de tipo matriz. La información sobre estas *features* se almacena en un objeto DataFrame. Cada fila del DataFrame proporciona información sobre las *features*. Las columnas del DataFrame representan diferentes atributos de las *features* de interés, p. ej., identificadores de genes o transcritos, etc.

El objeto de la clase SummarizedExperiment contiene en este caso:

- Una matriz que contiene los datos ómicos cuantitativos (concentraciones de metabolitos en este caso), almacenados como un objeto de tipo matriz. Las características (metabolitos) se definen en las filas y las muestras en las columnas.
- Un *slot* de metadatos de muestra que contiene las covariables de la muestra, almacenada como un *dataframe*. Las filas de esta tabla representan muestras (las filas coinciden exactamente con las columnas de los datos de concentración).
- Un *slot* de metadatos de características que contiene las covariables de los metabolitos, almacenada como un *dataframe*. Las filas de este marco de datos coinciden exactamente con las filas de los datos de concentración.

```
library("SummarizedExperiment")

stopifnot(rownames(t_datafile_matrix_reduced) == metabolite_metadata$Name) # compruebo que los nombres coinciden
stopifnot(colnames(t_datafile_matrix_reduced) == sample_metadata$SampleID) # compruebo que los nombres coinciden
rownames(sample_metadata)=sample_metadata$SampleID
```

```
rownames(metabolite_metadata)=metabolite_metadata$Name
se <- SummarizedExperiment(assays = list(counts = t_datafile_matrix_reduced), colData =
sample_metadata, rowData = metabolite_metadata) # construyo el SummarizedExperiment
se  # visualizo el SummarizedExperiment
```

```
## class: SummarizedExperiment
## dim: 149 140
## metadata(0):
## assays(1): counts
## rownames(149): M1 M2 ... M148 M149
## rowData names(4): Name Label Perc_missing QC_RSD
## colnames(140): sample_1 sample_2 ... sample_139 sample_140
## colData names(3): SampleID SampleType Class
```

```
assay(se)[1:5,1:5]   # visualizo la matriz de concentraciones que forma parte del
SummarizedExperiment
```

```
##    sample_1 sample_2 sample_3 sample_4 sample_5
## M1     90.1     43.0    214.3     31.6     81.9
## M2    491.6    525.7  10703.2     59.7    258.7
## M3    202.9    130.2    104.7     86.4    315.1
## M4     35.0       NA     46.8     14.0      8.7
## M5    164.2    694.5    483.4     88.6    243.2
```

4.2. Limpieza de datos

A continuación, realizo una limpieza de los datos. Es recomendable evaluar la calidad de los datos y eliminar (limpiar) cualquier metabolito con mediciones incorrectas antes de realizar cualquier análisis estadístico. Para el conjunto de datos utilizado en este ejemplo solo conservaré los metabolitos que cumplen los siguientes criterios:

- QC_RSD inferior al 20 %
- Ningún valor faltante

Una vez depurados los datos, se informa a través del SummarizedExperiment el número de metabolitos restantes, que son 21.

```
se1 <- se[rowData(se)$Perc_missing==0 & rowData(se)$QC_RSD<20,] # elimino los metabolitos según
los criterios descritos anteriormente
se1   # visualizo el nuevo SummarizedExperiment
```

```
## class: SummarizedExperiment
## dim: 21 140
## metadata(0):
## assays(1): counts
## rownames(21): M8 M15 ... M144 M149
## rowData names(4): Name Label Perc_missing QC_RSD
## colnames(140): sample_1 sample_2 ... sample_139 sample_140
## colData names(3): SampleID SampleType Class
```

```
assay(se1)[1:5, 1:8]
```

```
##     sample_1 sample_2 sample_3 sample_4 sample_5 sample_6 sample_7 sample_8
## M8      46.5    125.7     85.1     23.9     61.1    243.7     51.3     37.1
## M15     28.8    210.7     45.4     38.3     51.0     76.8     58.1     19.2
## M25     21.4     28.3     35.1     26.6     58.9     29.1     57.7      6.6
## M32     97.8    621.2    360.1    111.6    233.6     76.4    123.7     38.7
## M33    274.1    776.7    532.3    133.4    328.4    297.9    269.5    249.8
```

4.3. Análisis exploratorio de datos

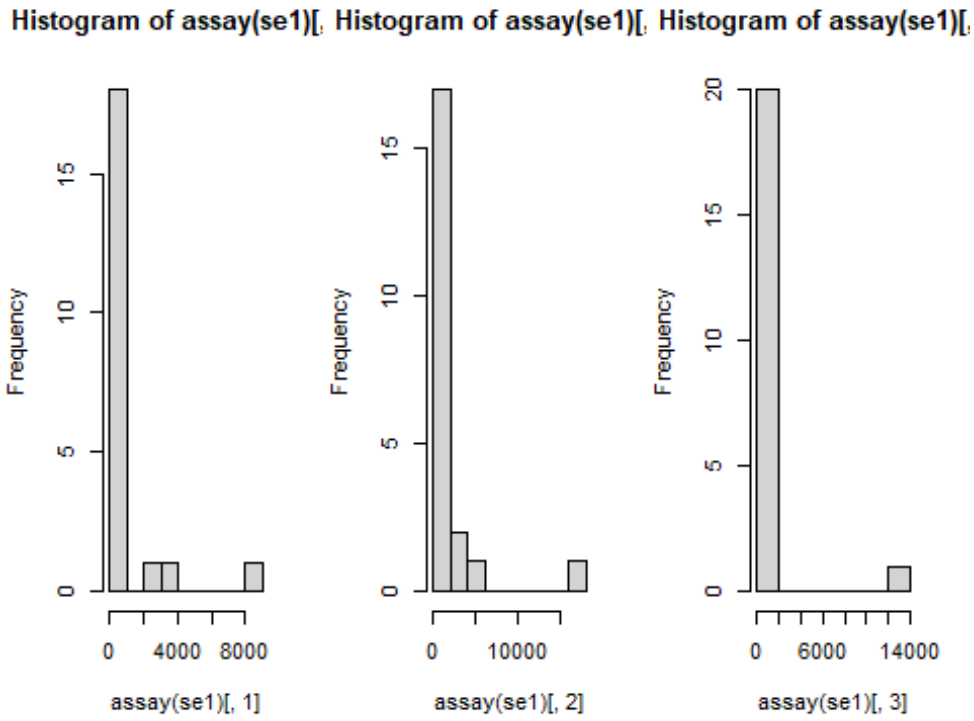
4.3.1. Análisis estadístico univariante

Estos datos de metabolómica son de alto rendimiento con grandes dimensiones, por lo que no es posible tener una buena visión general simplemente observando la matriz de datos. Se puede realizar una inspección inicial de los datos utilizando estadísticas de resumen básicas y gráficos básicos. Los histogramas nos permitirán ver la distribución de las concentraciones en una muestra en cada gráfico, como se muestra a continuación:

```
apply(assay(se1),2, summary)[1:5,1:8]  # Column-wise summary statistics
```

```
##         sample_1 sample_2 sample_3 sample_4 sample_5 sample_6 sample_7 sample_8
## Min.     21.4000     28.3   21.700  16.1000  12.2000  28.1000   22.300   6.6000
## 1st Qu.  33.7000    209.7   85.100  26.3000  61.1000  76.4000   57.700  29.7000
## Median  128.5000    403.3  277.900  91.6000 233.6000 207.2000  151.700  38.7000
## Mean    774.5619   1719.2 1044.576 415.4571 982.1286 819.2619 1071.186 325.2143
## 3rd Qu. 274.1000    961.8  689.400 160.1000 503.2000 850.6000  549.500 159.6000
```

```
opt <- par(mfrow=c(1,3))
hist(assay(se1)[,1]) # visualizo los histogramas
hist(assay(se1)[,2])
hist(assay(se1)[,3])
```

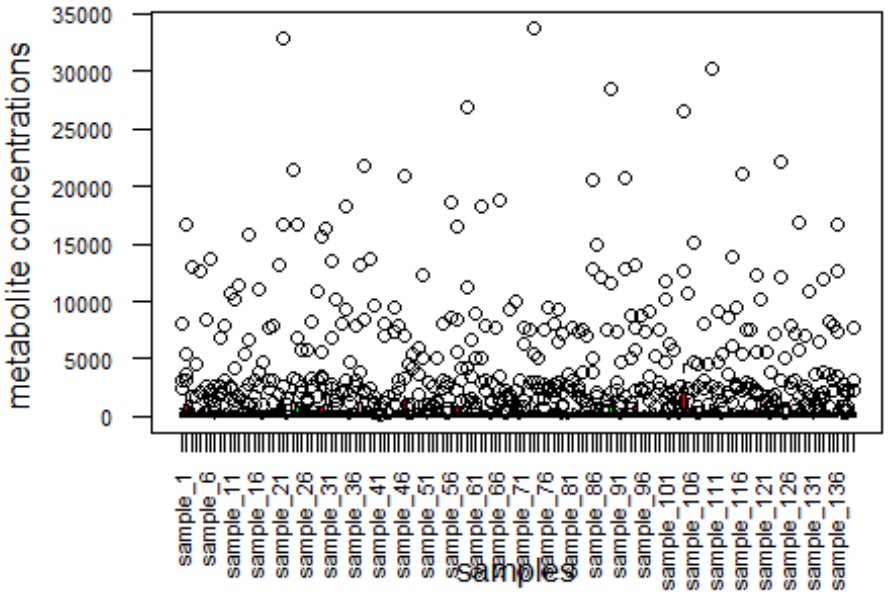



```
par(opt)
```

He observado que los datos tienen unidades de medida bastante distintas para cada metabolito. Ahora visualizo todas las muestras en un boxplot. Los boxplot permiten ver todas las muestras a la vez y así podré compararlas todas en un mismo gráfico para deducir si es necesario aplicar algún tipo de preprocesamiento:

```
colores <- c()  # asigno a cada muestra un color distinto para la visualización en los gráficos según el tipo de muestra
for (i in 1: 140) {
  if (colData(se1)$Class[i] =="QC"){colores<-c(colores,"blue")}
  if (colData(se1)$Class[i] =="GC"){colores<-c(colores,"red")}
  if (colData(se1)$Class[i] =="BN"){colores<-c(colores,"pink")}
  if (colData(se1)$Class[i] =="HE"){colores<-c(colores,"green")}
```

```
}
boxplot(assay(se1), col=colores, xlab="samples", ylab="metabolite concentrations ", las=2,
cex.axis=0.7, cex.main=0.7)
```

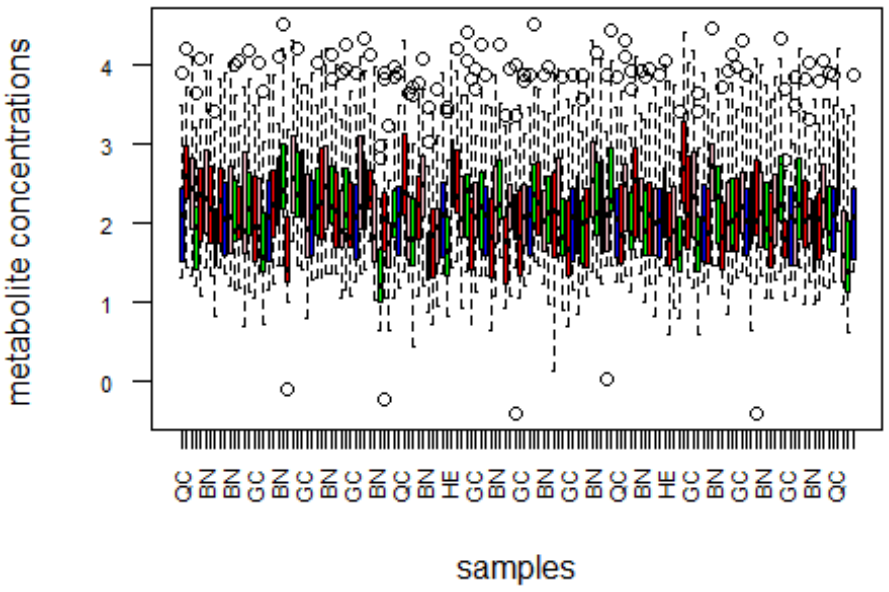



```
rownames(colData(se1))=colData(se1)$Class  # cambio los nombres de las filas a los tipos de
muestra para diferenciarlas según el tipo de muestra
```

Una vez he cambiado los nombres de las filas a los tipos de muestra para diferenciarlas según el tipo de muestra, procedo a transformar los datos con logaritmo en base 10 porque están en unidades de medida muy diferentes. Visualizo a continuación todas las muestras ya transformadas en un gráfico:

```
logX <- log10(assay(se1))  # Log scale (base-10)

boxplot(logX, col=colores, xlab="samples", ylab="metabolite concentrations ", las=2,
cex.axis=0.7, cex.main=0.7)
```



Se puede apreciar en el gráfico anterior con todas las muestras que las muestras rojas (GC) parecen tener valores de concentraciones más extremas que las demás muestras. Las muestras de QC parece que tienen los valores más igualados centrados en el 2. Las muestras de BN y HE parece que están también más centradas en el 2 aunque no tan igualadas como las de QC.

4.3.2. Visualización de datos multivariante

4.3.2.1. Calculo de las componentes principales (PCA)

Para proporcionar una evaluación multivariante de la calidad del conjunto de datos depurados, es recomendable realizar un análisis de componentes principales simple, tras la transformación y el escalado adecuados. El gráfico del PCA se etiquetará por tipo de muestra, es decir, control de calidad (QC) o muestra biológica. Los datos de alta calidad tendrán QC que se agrupan estrechamente en comparación con las muestras biológicas.

Los valores ya se habían transformado logarítmicamente anteriormente, entonces para el cálculo del PCA escalaré los datos transformados logarítmicamente.

Se obtiene un gráfico muy útil calculando las componentes principales y representando gráficamente las dos primeras componentes. Esto puede utilizarse para detectar muestras inusuales o efectos de lote. Con este método se reduce la dimensionalidad del conjunto de datos para encontrar los factores que realmente explican la variabilidad de los datos [3].

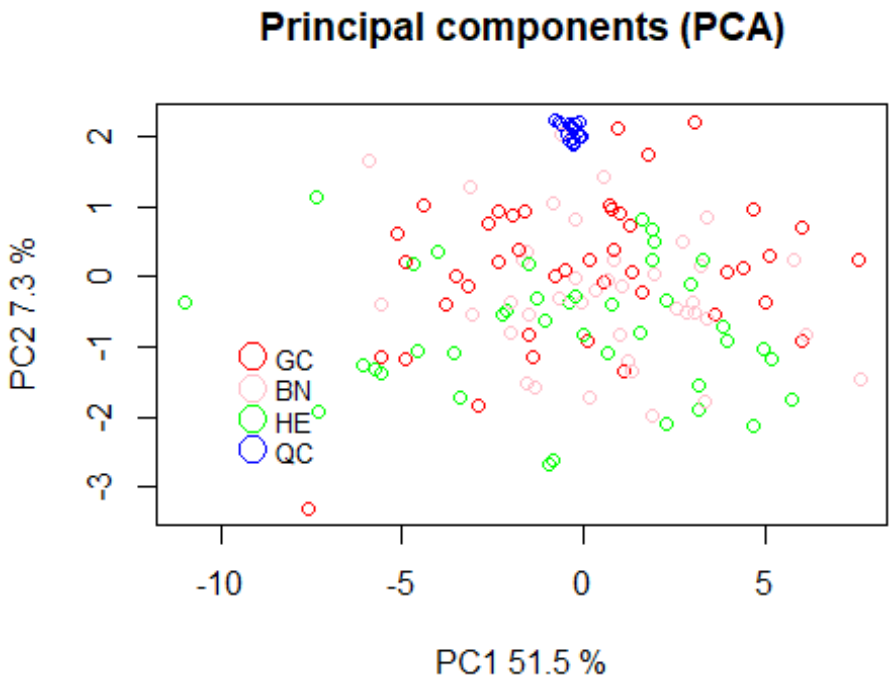
Primero, calculo las componentes principales y los *loadings*:

```
pcX<-prcomp(t(logX), center = TRUE, scale = TRUE) #  escalando y centrando los datos
loads<- round(pcX$sdev^2/sum(pcX$sdev^2)*100,1)
```

A continuación, represento las dos primeras componentes:

```
xlab<-c(paste("PC1",loads[1],"%"))
ylab<-c(paste("PC2",loads[2],"%"))

plot(pcX$x[,1:2],xlab=xlab,ylab=ylab, col=colores, main ="Principal components (PCA)")
legend("bottomleft",
       legend = c("GC", "BN", "HE", "QC"),
       col = c("red", "pink", "green", "blue"),
       pch = c(1,1),
       bty = "n",
       pt.cex = 2,
       cex = 0.8,
       text.col = "black",
       horiz = F ,
       inset = c(0.1, 0.1))
```



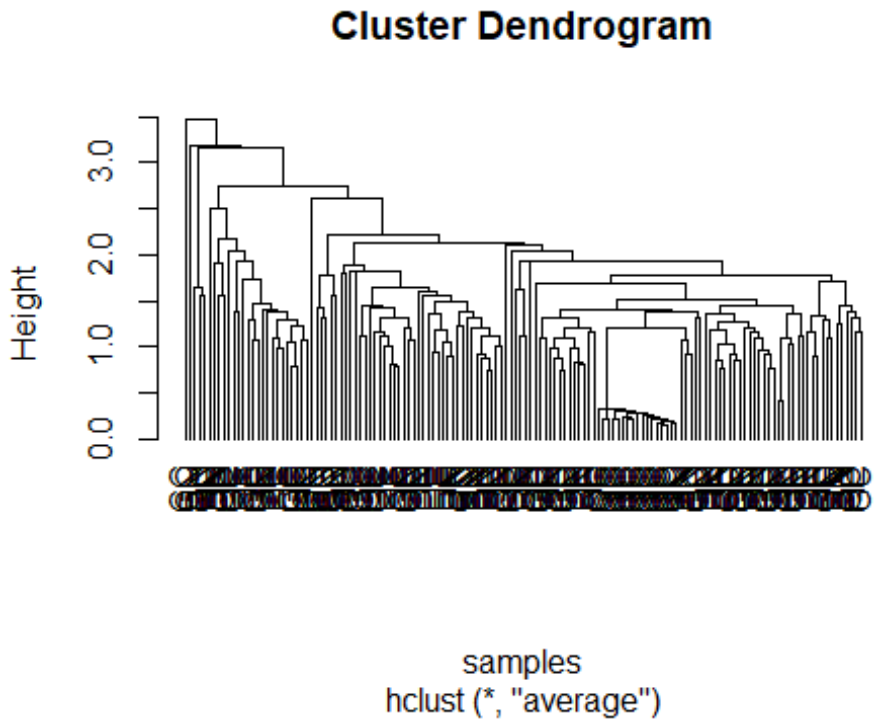
Se observa claramente como las muestras de QC están muy juntas todas en el gráfico, mientras que las demás están dispersas en el gráfico. Esto quiere decir que estas muestras de control de calidad son casi idénticas en cuanto a concentración de metabolitos, lo que era de esperar. Se confirma entonces que los datos son de alta calidad porque los QC se agrupan estrechamente en comparación con las muestras biológicas.

4.3.2.2. Técnicas de clustering por Agrupamiento jerárquico

Las técnicas de clustering consisten en formar grupos, lo más homogéneos posible, a partir de un conjunto de observaciones que tienen características similares para así crear patrones y los datos que presentan características similares puedan clasificarse en sus grupos objetivo correspondientes. Los métodos más utilizados pueden ser de tipo jerárquico o no jerárquico [3]. Los métodos jerárquicos crean grupos buscando el número de clústeres óptimo en cada momento, sin determinarse previamente por el usuario [3].

```
# Realizamos agrupamiento jerárquico

clust.euclid.average <- hclust(dist(t(logX)),method="average") # Calcula la matriz de distancias
y realiza el agrupamiento
plot(clust.euclid.average, hang=-1, xlab = "samples") # Representamos el dendrograma
```



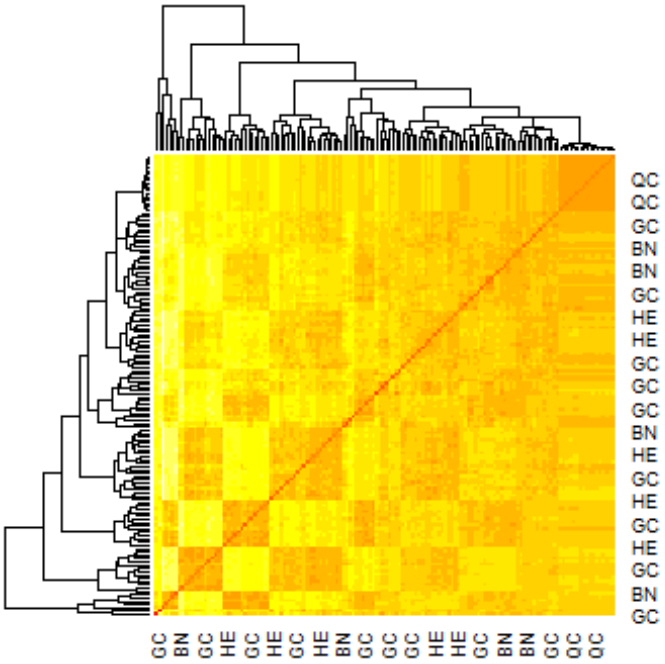
Se observa de nuevo que las muestras QC se agrupan estrechamente en el dendrograma en comparación con las muestras biológicas, por lo cual se concluye que los datos son de alta calidad.

Tanto el PCA como la agrupación en clústeres muestran una agrupación clara asociada con el tipo de muestra QC, lo que es bastante esperable, excepto si hay problemas inesperados, como muestras problemáticas o efectos de lote. Se confirma entonces que los datos son de alta calidad porque los QC se agrupan estrechamente en comparación con las muestras biológicas.

4.3.2.3. Análisis basado en distancias

Otro método para visualizar distancias entre muestras es realizar un análisis basado en distancias. Podemos realizarlo calculando la matriz de distancias y visualizándola mediante la generación de un *heatmap* (mapa de colores):

```
manDist <- dist(t(logX))
heatmap (as.matrix(manDist), col=heat.colors(16))
```



Se observa de nuevo que los datos son de alta calidad porque los QC se agrupan estrechamente en el dendrograma vertical y horizontal en comparación con las muestras biológicas. Esta agrupación en clúster de las muestras QC se observa como un cuadrado naranja más intenso en la esquina superior derecha del mapa de colores.

Todas las visualizaciones coinciden en mostrar una separación asociada al factor muestra control de calidad QC, lo que confirma que los datos son de alta calidad.

5. Discusión

Se pudo apreciar en el gráfico boxplot con todas las muestras que las muestras rojas (GC) parecen tener valores de concentraciones más extremas que las demás muestras. Todas las visualizaciones coincidieron en mostrar una separación asociada al factor muestra control de calidad QC, lo que confirma que los datos son de alta calidad.

En cuanto a las limitaciones del análisis, hay que tener en cuenta que algunos metabolitos utilizados no son específicos de cáncer gástrico y pueden confundir la interpretación. Por ejemplo, los niveles elevados de alanina en la orina de pacientes con GC en comparación con HE muestran que la alanina puede ser un biomarcador de desgaste muscular, pero no necesariamente un biomarcador específico de cáncer [1].

Otra limitación es que el estudio original emparejó a los pacientes según tres factores de confusión comunes: edad, sexo e IMC. Sin embargo, al ser un diseño observacional, solo se pudieron controlar los factores de confusión conocidos. Otros factores de confusión no incluidos en el estudio podrían alterar la interpretación [1].

6. Conclusiones

El análisis exploratorio de los datos mostró que los datos son de alta calidad.

7. Referencias

[1] 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

[2] Nistal-Nuno-Beatriz-PEC1. https://github.com/datascience100/Nistal-Nuno-Beatriz-PEC1

[3] Casals, M. y Vila A. (2024). Introducción al machine learning R. [Recurso de aprendizaje textual]. 1.a ed. Fundació Universitat Oberta de Catalunya (FUOC).