Análisis Metadatos-Caquexia

Adriana Pastor

2025-04-01

INTRODUCCIÓN

El objetivo de esta actividad es realizar un análisis exploratorio de muestras de pacientes con caquexia y controles, con el fin de analizar la expresión de distintos metabolitos.

En primer lugar, se creará el objeto *SummarizedExperiment* para trabajar con los datos. El beneficio de utilizar este tipo de objeto es que, basándonos en un modelo, podemos separar los metadatos de los datos numéricos, lo que facilita el manejo de datos grandes y complejos de manera mucho más sencilla.

Es útil utilizar esta extensión de la clase *ExpressionSet*. En ambos casos, podemos manejar datos de alta dimensión, como es el caso en este estudio. *SummarizedExperiment* está diseñado de manera que resulta más flexible y general. Permite trabajar con cualquier tipo de datos, como metabolómicos, proteómicos, transcriptómicos, entre otros, y agregar diversos *assays* (tipos de datos) al mismo objeto.

Además, las diferencias en la forma en que se almacena la información permiten una mayor flexibilidad, sin limitarse a un solo tipo de datos o características.

MATERIALES Y MÉTODOS

Los datos utilizados se obtuvieron del repositorio de GitHub *Datasets/2024-Cachexia*: https://github.com/nutrimetabolomics/metaboData/tree/main/Datasets/2024-Cachexia.

Como metodología, realizaremos un **análisis de componentes principales (PCA)** y un **análisis de clustering**.

2.1 Carga y procesamiento de los datos

En primer lugar, cargamos los datos y convertimos la variable *Muscle loss* en un factor:

```
caquexia_1 <- read_csv("C:/Users/Adria/Desktop/ADO/PEC1/human_cachexia.csv") %>% as.data.frame
```

```
## 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.
```

```
str(caquexia_1)
```

```
## 'data.frame':    77 obs. of  65 variables:
##  $ Patient ID               : chr  "PIF_178" "PIF_087" "PIF_090" "NETL_005_V1" ...
##  $ Muscle loss              : chr  "cachexic" "cachexic" "cachexic" "cachexic" ...
##  $ 1,6-Anhydro-beta-D-glucose: num  40.9 62.2 270.4 154.5 22.2 ...
##  $ 1-Methylnicotinamide     : num  65.4 340.4 64.7 53 73.7 ...
```

```
##  $ 2-Aminobutyrate           : num  18.7 24.3 12.2 172.4 15.6 ...
##  $ 2-Hydroxyisobutyrate      : num  26.1 41.7 65.4 74.4 83.9 ...
##  $ 2-Oxoglutarate            : num  71.5 67.4 23.8 1199.9 33.1 ...
##  $ 3-Aminoisobutyrate        : num  1480.3 116.8 14.3 555.6 29.7 ...
##  $ 3-Hydroxybutyrate         : num  56.83 43.82 5.64 175.91 76.71 ...
##  $ 3-Hydroxyisovalerate      : num  10.1 79.8 23.3 25 69.4 ...
##  $ 3-Indoxylsulfate          : num  567 369 665 412 166 ...
##  $ 4-Hydroxyphenylacetate    : num  120.3 432.7 292.9 214.9 97.5 ...
##  $ Acetate                   : num  126.5 212.7 314.2 37.3 407.5 ...
##  $ Acetone                   : num  9.49 11.82 4.44 206.44 44.26 ...
##  $ Adipate                   : num  38.1 327 131.6 144 15 ...
##  $ Alanine                   : num  314 871 464 590 1119 ...
##  $ Asparagine                : num  159.2 157.6 89.1 273.1 42.5 ...
##  $ Betaine                   : num  110 245 117 279 392 ...
##  $ Carnitine                 : num  265.1 120.3 25 200.3 84.8 ...
##  $ Citrate                   : num  3714 2618 863 13630 854 ...
##  $ Creatine                  : num  196.4 212.7 221.4 85.6 105.6 ...
##  $ Creatinine                : num  16482 15835 24588 20952 6768 ...
##  $ Dimethylamine             : num  633 608 735 1064 242 ...
##  $ Ethanolamine              : num  645 488 407 821 365 ...
##  $ Formate                   : num  441 252 250 469 114 ...
##  $ Fucose                    : num  337 198.3 186.8 407.5 26.1 ...
##  $ Fumarate                  : num  7.69 18.92 7.1 96.54 19.69 ...
##  $ Glucose                   : num  395 8691 1353 863 6836 ...
##  $ Glutamine                 : num  871 602 302 1686 433 ...
##  $ Glycine                   : num  2039 1108 620 5064 395 ...
##  $ Glycolate                 : num  685.4 652 141.2 70.8 26.6 ...
##  $ Guanidoacetate            : num  154 110 183 103 53 ...
##  $ Hippurate                 : num  4582 1737 4316 757 1153 ...
##  $ Histidine                 : num  925 846 284 1043 327 ...
##  $ Hypoxanthine              : num  97.5 82.3 114.4 223.6 66.7 ...
##  $ Isoleucine                : num  5.58 8.17 9.3 37.71 40.04 ...
##  $ Lactate                   : num  107 369 750 369 3641 ...
##  $ Leucine                   : num  42.1 77.5 31.5 103.5 101.5 ...
##  $ Lysine                    : num  146.9 284.3 97.5 290 122.7 ...
##  $ Methylamine               : num  52.5 23.6 18.7 48.9 27.9 ...
##  $ Methylguanidine           : num  9.97 7.69 4.66 141.17 5.31 ...
##  $ N,N-Dimethylglycine       : num  23.3 87.4 24.5 40 46.1 ...
##  $ O-Acetylcarnitine         : num  52.98 50.4 5.58 254.68 45.6 ...
##  $ Pantothenate              : num  25.8 186.8 145.5 42.5 74.4 ...
##  $ Pyroglutamate             : num  437 437 713 567 185 ...
##  $ Pyruvate                  : num  21.1 37 29.4 64.1 12.3 ...
##  $ Quinolinate               : num  165.7 73 192.5 86.5 38.1 ...
##  $ Serine                    : num  284 392 296 1249 206 ...
##  $ Succinate                 : num  154.5 244.7 142.6 144 68.7 ...
##  $ Sucrose                   : num  45.1 459.4 160.8 111 75.2 ...
##  $ Tartrate                  : num  97.51 32.79 16.28 837.15 4.53 ...
##  $ Taurine                   : num  1920 1261 4273 1525 469 ...
##  $ Threonine                 : num  184.9 198.3 110 376.1 64.1 ...
##  $ Trigonelline              : num  943.9 208.5 192.5 992.3 86.5 ...
##  $ Trimethylamine N-oxide    : num  2122 639 1153 1451 172 ...
##  $ Tryptophan                : num  259.8 83.1 82.3 235.1 103.5 ...
##  $ Tyrosine                  : num  290 167.3 60.3 323.8 142.6 ...
##  $ Uracil                    : num  111 47 31.5 30.6 44.3 ...
```

```
##  $ Valine                    : num  86.5 110 59.1 102.5 160.8 ...
##  $ Xylose                    : num  72.2 192.5 2164.6 125.2 186.8 ...
##  $ cis-Aconitate             : num  237 334 330 1863 101 ...
##  $ myo-Inositol              : num  135.6 376.1 86.5 247.2 750 ...
##  $ trans-Aconitate           : num  51.9 217 58.6 75.9 98.5 ...
##  $ pi-Methylhistidine        : num  157.6 308 145.5 249.6 84.8 ...
##  $ tau-Methylhistidine       : num  160.8 130.3 83.9 254.7 79.8 ...
```

```r
caquexia_1$`Muscle loss` <- factor(caquexia_1$`Muscle loss`) #Convertimos a factor
```

A continuación, extraemos las columnas de mediciones, manteniendo únicamente aquellas que contienen datos numéricos de interés. También creamos un nuevo *data frame* con las columnas *Patient ID* y *Muscle loss*, que utilizaremos como metadatos:

```r
# Primero, seleccionamos las columnas numéricas de interés
exp_data <- caquexia_1[, -c(1, 2)]

# Asignar los nombres de las filas
rownames(exp_data) <- caquexia_1$`Patient ID`


# Crear un nuevo DataFrame extrayendo dos columnas
colData <- caquexia_1[, c("Patient ID", "Muscle loss")]
```

Dado que los datos generalmente deben estar organizados con las muestras en las filas y las variables (metabolitos, genes, etc.) en las columnas, es necesario transponer la matriz de datos antes de realizar el Análisis de Componentes Principales (PCA).

```r
# traslocamos para que cuadren las filas y las coumnas
exp_data_t <- t(as.matrix(exp_data))
```

Finalmente, organizamos los datos en un objeto `SummarizedExperiment`, lo que nos permite almacenar conjuntamente las mediciones y los metadatos, facilitando su análisis en estudios de expresión génica o metabólica.

```r
# Crear el objeto SummarizedExperiment
se <- SummarizedExperiment(assays = list(counts = exp_data_t), colData = colData)

# Verificar la estructura del objeto SummarizedExperiment
se
```

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

2.2 Análisis de Componentes Principales (PCA)

Existe una variación muy amplia en el rango de valores de las variables, como podemos observar. Para homogeneizar la escala de los datos, aplicamos una transformación logarítmica antes de realizar el **Análisis de Componentes Principales (PCA)** utilizando la función `prcomp()`.

Esto permite minimizar la influencia de los valores extremos y mejorar la interpretación de los componentes

principales.

```
summary(as.vector(t(assays(se)$counts)))
```

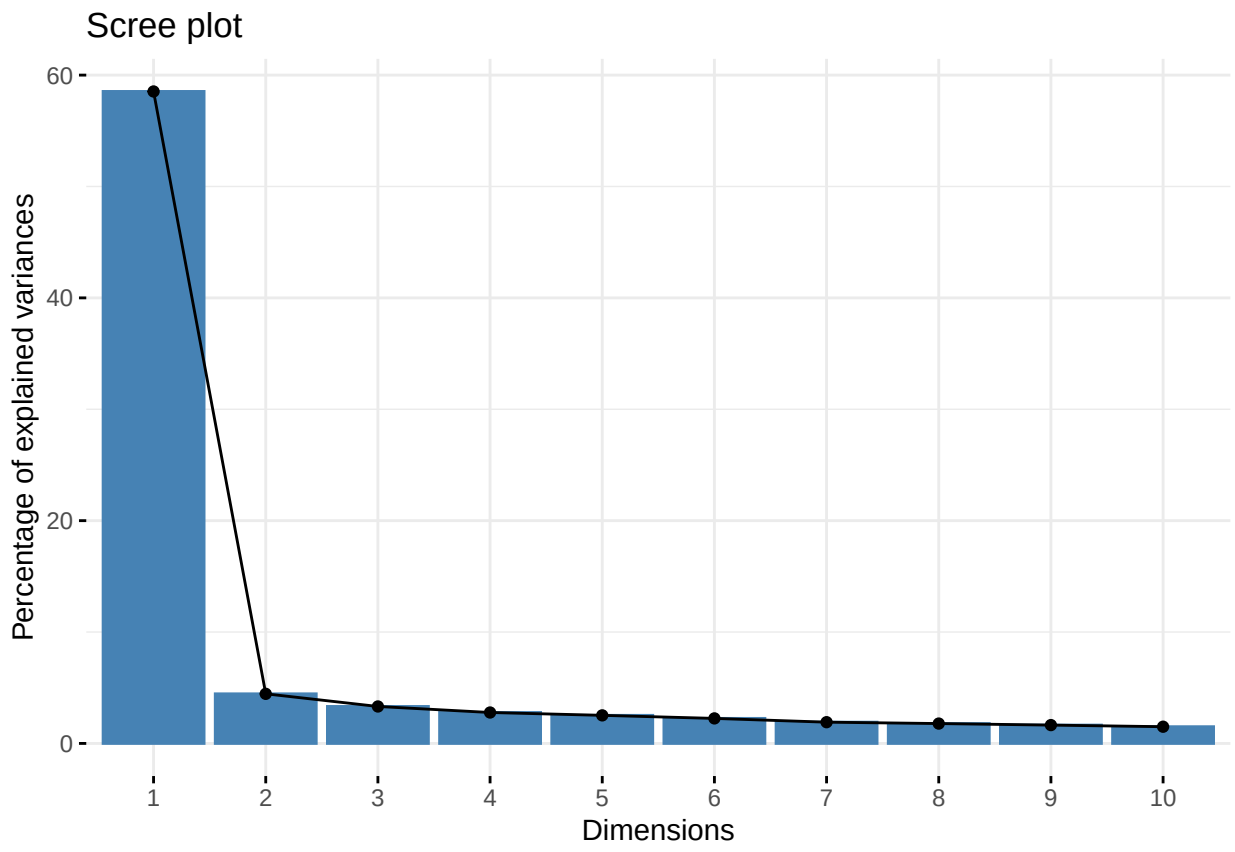
```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     0.79    17.46    51.42   347.37   160.77 33860.35
```

```
# Realizar el PCA
pca_result <- prcomp(log(t(assays(se)$counts) + 1), scale=TRUE)
```

Para determinar cuántos componentes principales son relevantes para nuestro análisis, realizamos un gráfico de la **varianza explicada acumulada**, también conocido como **método del codo**.

Este gráfico nos permite identificar el punto en el que agregar más componentes no aporta una ganancia significativa en la variabilidad explicada. En nuestro caso, observamos que los **componentes principales 1 y 2** capturan la mayor parte de la variabilidad de los datos, por lo que nos enfocaremos en ellos para continuar con el análisis.

```
fviz_eig(pca_result)
```

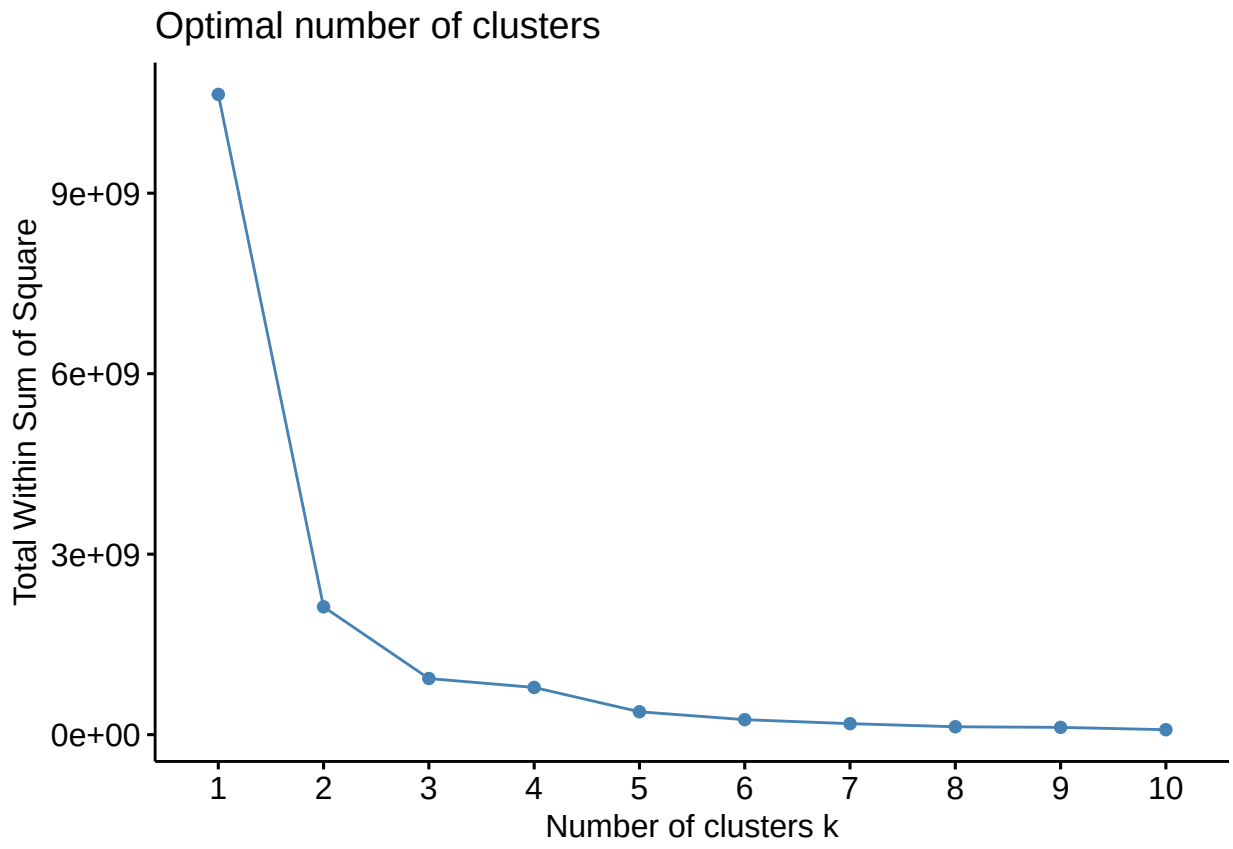


Este resultado justifica la selección de los **primeros dos componentes principales (PC1 y PC2)** para la visualización y análisis posterior.

```
# Seleccionar los primeros 2 componentes principales
dat.sel <- pca_result$x[, 1:2]
```

2.3 Diferenciación de grupos por Clustering

```
#definimos el numero de clusters idoneo mediante el metodo del codo
fviz_nbclust(assays(se)$counts, kmeans, method = "wss")
```




```
# Definir el número de clusters segun el metodo anterior
k <- 2
```

Para determinar el número óptimo de **clusters** en los que debemos agrupar nuestras muestras, utilizamos el **método del codo**. Este método se basa en calcular la suma de los cuadrados dentro de los grupos (**within-cluster sum of squares, WSS**) para diferentes valores de **k** y graficar el resultado.

El número óptimo de clusters se identifica en el **"codo"** de la curva, es decir, el punto donde agregar más clusters deja de reducir significativamente la variabilidad dentro de los grupos.

En nuestro caso, observamos que $\mathbf{k = 2}$ es el valor más adecuado.

```
set.seed(123)  # Fijar semilla para reproducibilidad
km <- kmeans(dat.sel, centers = k, iter.max = 1000)
```

```
# Agregamos la información del cluster a los datos de PCA
pca_cluster_df <- data.frame(dat.sel, Cluster = factor(km$cluster), Grupo = factor(se$`Muscle loss`))
```

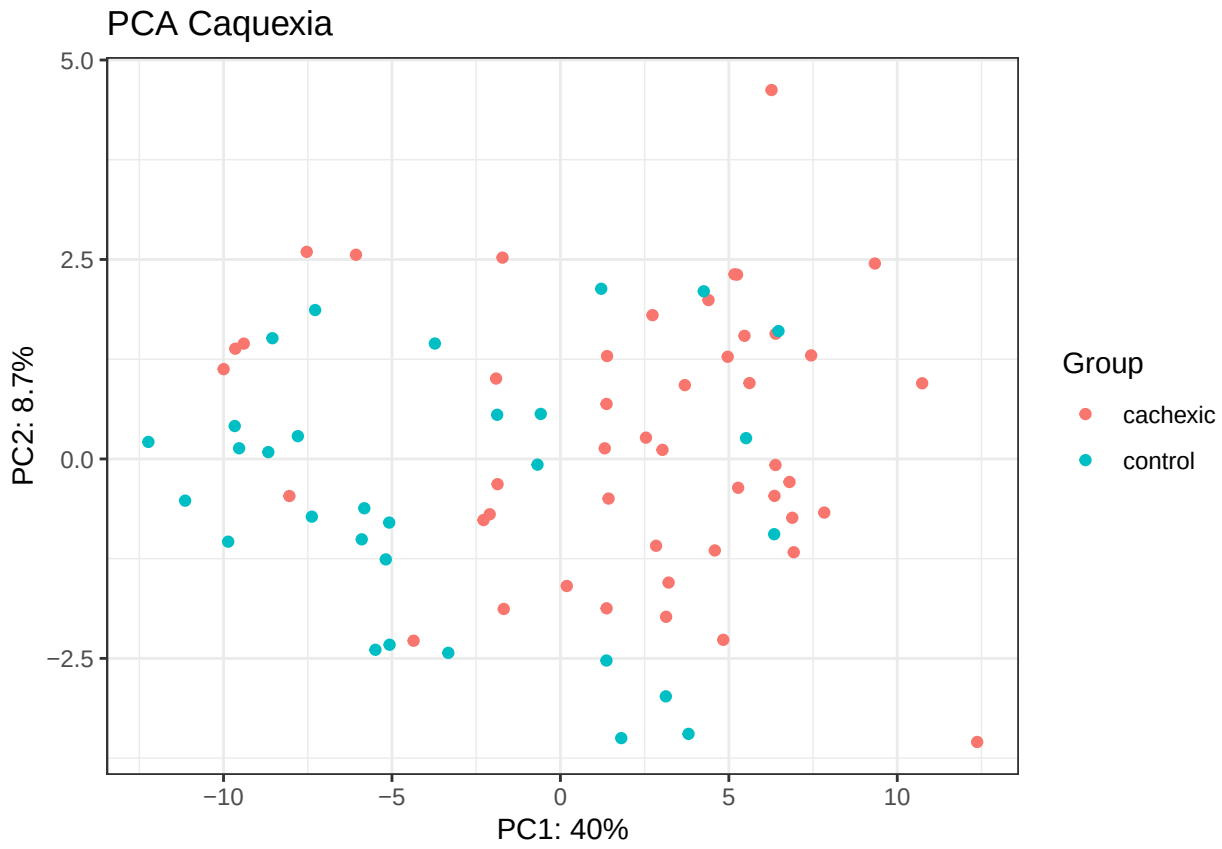
RESULTADOS

3.1 Los dos primeros componentes principales explican la mayor parte de la variabilidad de nuestros datos

```
pc1 <- pca_result$x[,1]
pc2 <- pca_result$x[,2]
group <- se$`Muscle loss`

pca_df <- data.frame("PC1" = as.vector(pc1),
                     "PC2" = as.vector(pc2),
                     "Group" = group)

p <- ggplot(pca_df, aes(x=PC1, y=PC2)) +
  geom_point(aes(color=Group)) +
  # scale_color_manual(c("orange", "blue")) +
  ggtitle("PCA Caquexia") +
  labs(x="PC1: 40%",
       y = "PC2: 8.7%") +
  theme_bw()
p
```



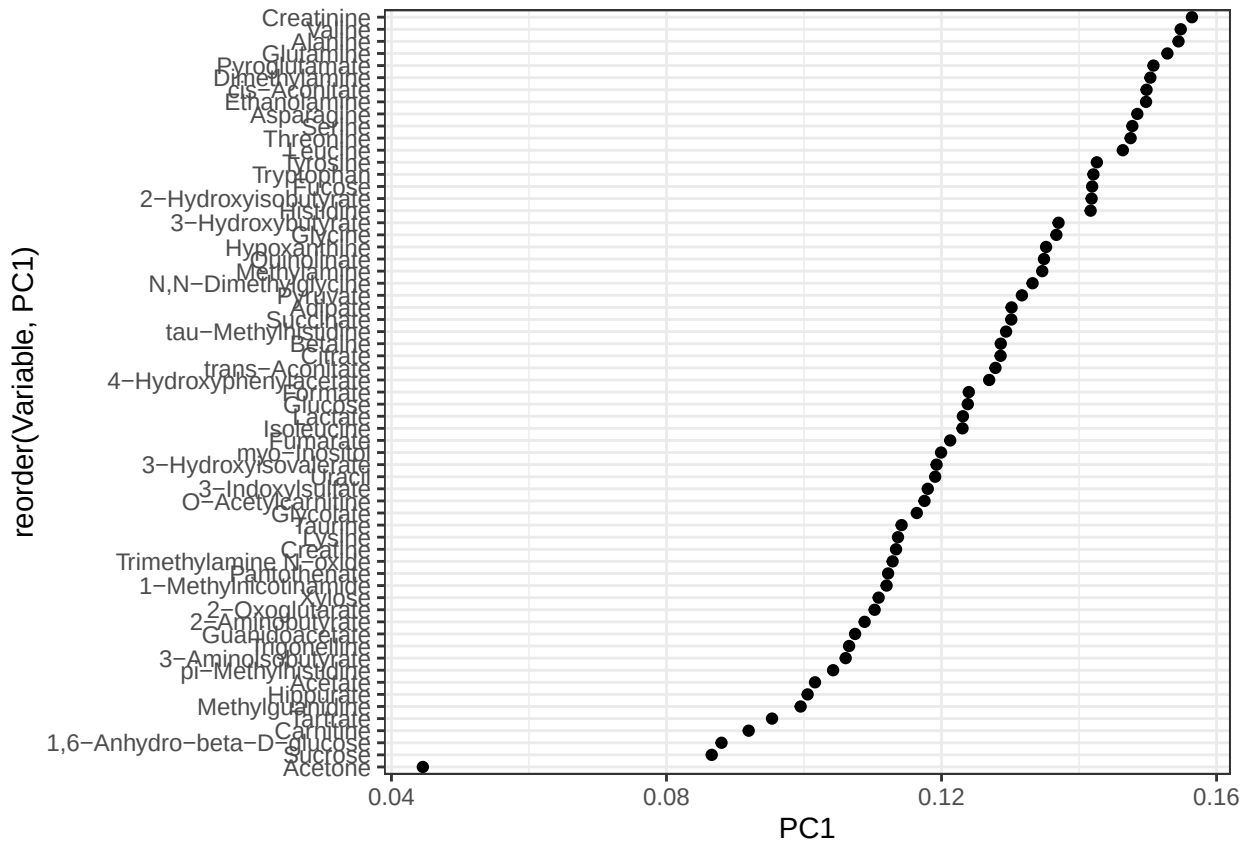
Mostramos en una gráfica de dispersión los dos primeros componentes principales y analizamos cómo se distribuyen los sujetos a lo largo de estos ejes. Diferenciamos a los individuos en función de los dos grupos que tenemos: **control** y **caquexia**.

Podemos observar una cierta tendencia hacia la derecha, lo que sugiere que existe una diferencia significativa

entre los grupos. Sin embargo, las variables elegidas para estos componentes no parecen separar completamente ambas condiciones.

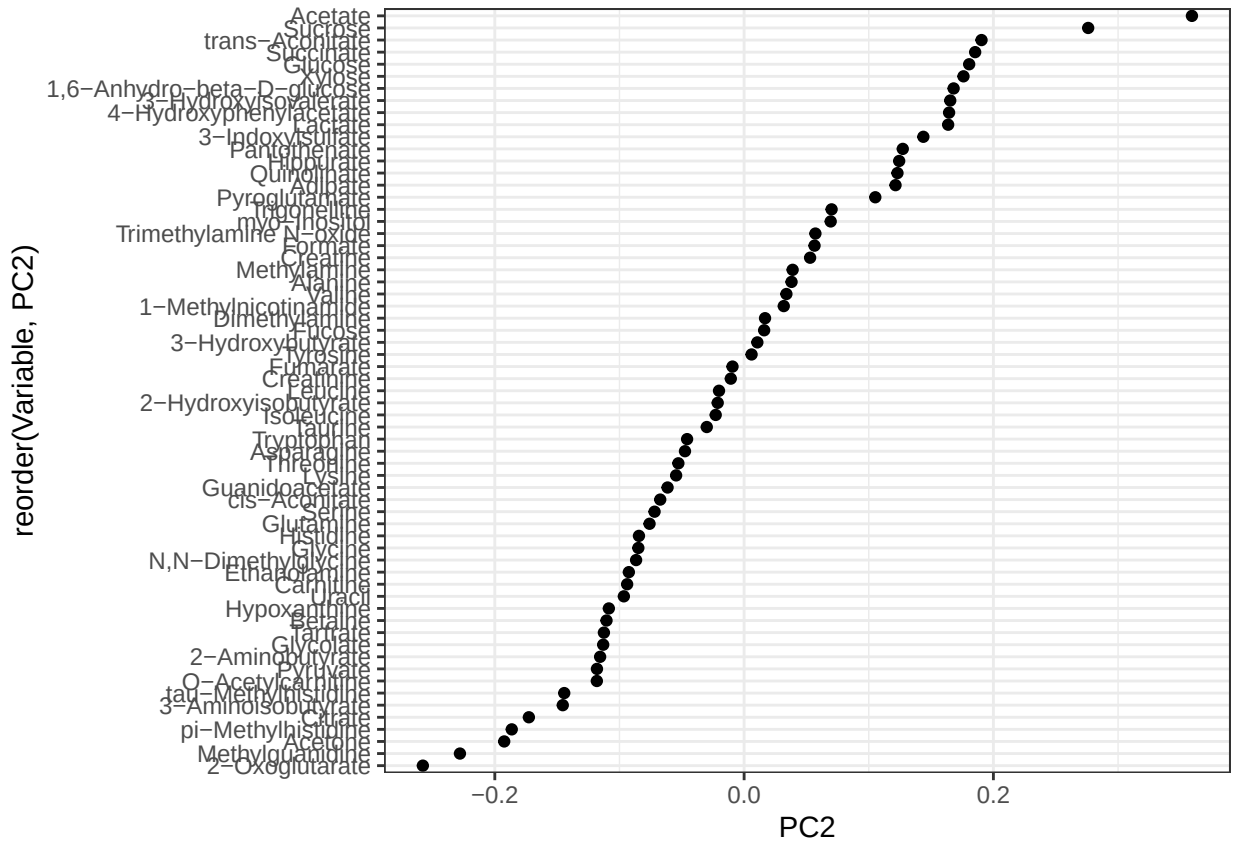
Analizando la contribución de las variables en cada componente principal, observamos que la creatinina tiene un alto peso en PC1, lo que indica que influye significativamente en la variabilidad de este componente. En contraste, la acetona tiene un impacto mínimo en PC1.

```
pca_result$rotation %>%
  as.data.frame() %>%
  mutate(Variable = row.names(.)) %>%
  ggplot(aes(x=PC1, y=reorder(Variable, PC1))) +
  geom_point() +
  theme_bw()
```



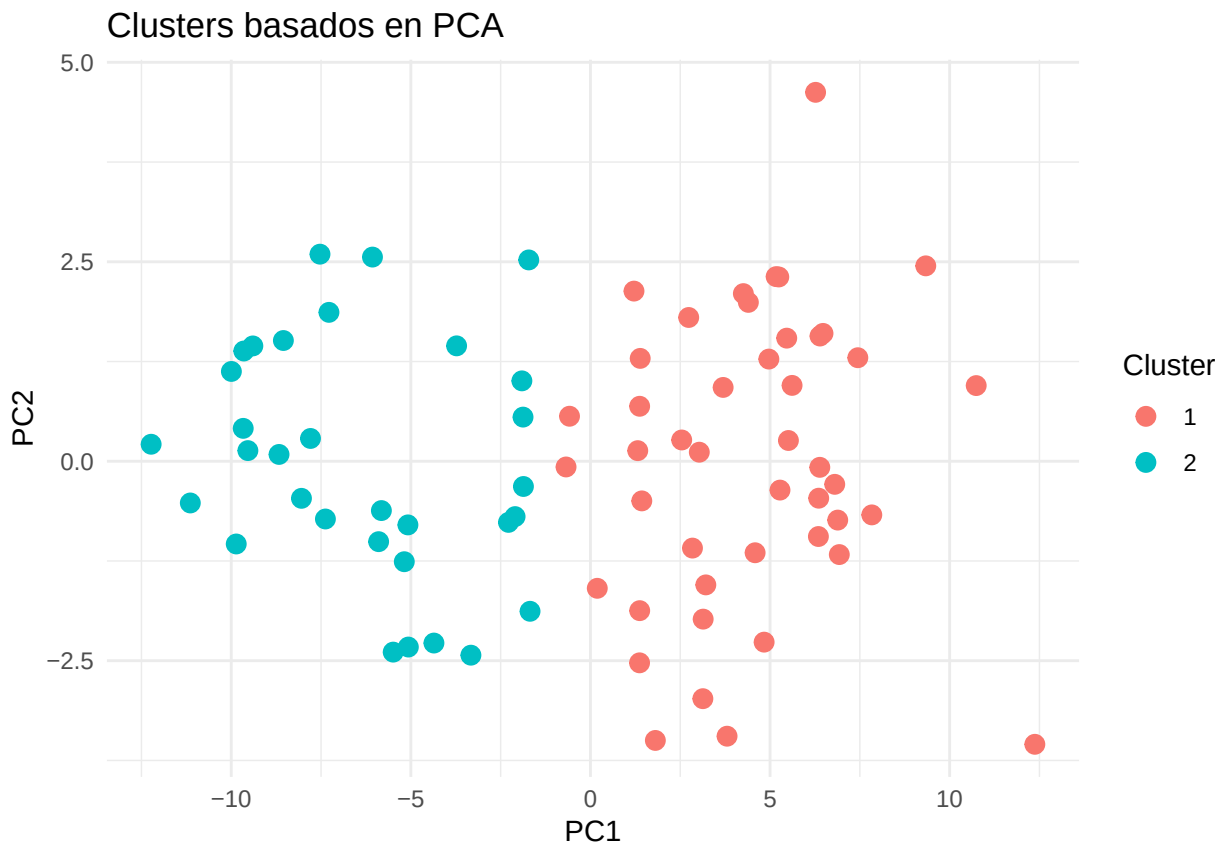
Para PC2, el acetato presenta una contribución elevada, mientras que el 2-oxoglutarato apenas afecta a este componente

```
pca_result$rotation %>%
  as.data.frame() %>%
  mutate(Variable = row.names(.)) %>%
  ggplot(aes(x=PC2, y=reorder(Variable, PC2))) +
  geom_point() +
  theme_bw()
```

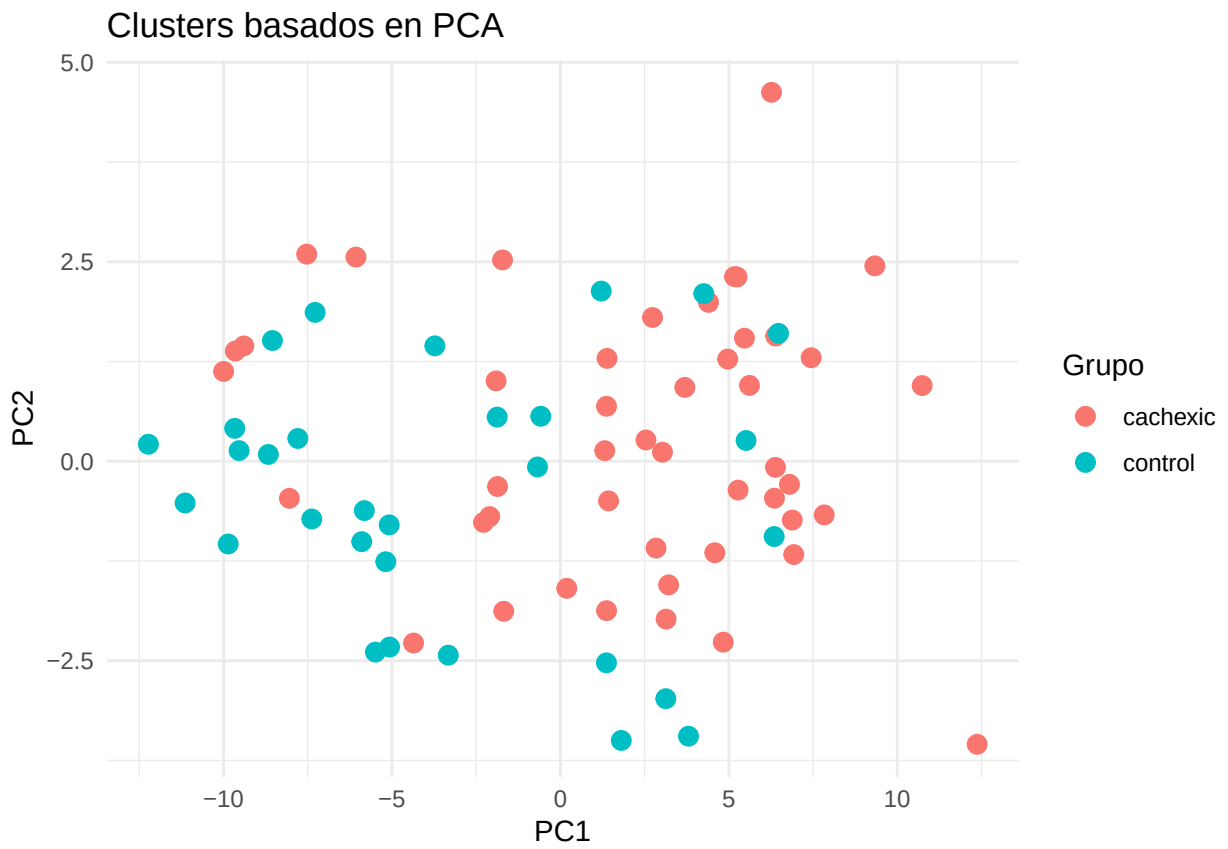


3.2 El metodo de Clsuterin g k-means aporta diferencia entre la diferencia entre datos

```
# Graficar los clusters en un scatterplot con ggplot2
ggplot(pca_cluster_df, aes(x = PC1, y = PC2, color = Cluster)) +
  geom_point(size = 3) +
  labs(title = "Clusters basados en PCA", x = "PC1", y = "PC2") +
  theme_minimal()
```




```
ggplot(pca_cluster_df, aes(x = PC1, y = PC2, color = Grupo)) +
  geom_point(size = 3) +
  labs(title = "Clusters basados en PCA", x = "PC1", y = "PC2") +
  theme_minimal()
```



Generamos una gráfica donde los sujetos se distribuyen en función de los dos primeros componentes principales y se agrupan en dos clusters.

Observamos una separación notable en dos grupos, aunque esta no coincide exactamente con la clasificación basada en la presencia de la enfermedad. Sin embargo, sí se detecta una tendencia en la que los sujetos control se asocian más con la parte izquierda del gráfico, mientras que los pacientes con caquexia tienden a ubicarse más a la derecha.

DISCUSIÓN

Aunque hemos identificado dos componentes principales que explican una proporción significativa de la variabilidad en los datos, los resultados obtenidos no son completamente concluyentes. En cuanto a los clusters, aunque se observa una tendencia en el primer componente que sugiere una cierta diferencia entre los grupos, la separación no es absoluta. Esto indica que hay otros factores o componentes que también podrían estar influyendo en la variabilidad de los datos, más allá de la clasificación simple en caquexia y control.

Aunque se puede percibir una cierta distinción entre los grupos de controles y pacientes con caquexia, esta no es tan clara ni definitiva. Para obtener conclusiones más robustas, sería necesario llevar a cabo un análisis más exhaustivo, considerando factores adicionales como el sexo, el estilo de vida, la presencia de otras patologías, etc.

Estos hallazgos sugieren que, aunque los componentes principales pueden capturar algunas diferencias entre los grupos, esas diferencias no se alinean de manera completamente clara con la clasificación clínica. Esto subraya la complejidad de los datos y la necesidad de análisis adicionales para comprender mejor las causas subyacentes de la variabilidad observada.

REFERENCIAS

https://github.com/adrianapastor1/PASTOR-GARCIA-ADRIANA-PEC1