ADO PEC1

Jesús Cea García

2025-03-31

1. Abstract

En este trabajo se construyó un objeto *SummarizedExperiment* a partir de datos de metabolómica en orina de pacientes caquéxicos y controles. Se exploraron las concentraciones de metabolitos por grupo, observando diferencias claras entre ambos. Un análisis de componentes principales (PCA) evidenció una separación más o menos marcada entre caquéxicos y controles, sin mostrar *batch effect* asociado al tipo de cáncer. Además, un dendrograma confirmó esta separación a nivel jerárquico. Finalmente, se realizó un análisis de expresión diferencial y un volcano plot y se identificaron metabolitos con diferencias significativas de concentración entre los grupos, los cuales pueden ser potencialmente relevantes como biomarcadores de caquexia.

2. Objetivos

Los objetivos de este trabajo se enumeran a continuación:

1. Integrar los datos metabolómicos de un archivo .csv, separando la información de concentraciones, metadatos y características de los metabolitos, para construir un objeto de clase *SummarizedExperiment*.
2. Explorar las diferencias globales en las concentraciones de metabolitos entre pacientes caquéxicos y controles.
3. Discriminar entre ambos grupos mediante técnicas de análisis multivariante.
4. Detectar y caracterizar posibles efectos de batch relacionados con el tipo de cáncer.
5. Identificar metabolitos diferencialmente expresados que puedan actuar como biomarcadores asociados a la caquexia

3. Métodos

3.1. Origen de los datos

Los datos utilizados provienen del repositorio habilitado para la PEC de la asignatura, los cuales parecen provenir del paquete de R specmine.datasets. Este dataset contiene datos provenientes del estudio de Eisner et al. (2010), en el que se analizaron perfiles de 63 metabolitos en muestras de orina (concentración en μM) de pacientes oncológicos mediante espectroscopía de resonancia magnética nuclear (1H-NMR). El estudio tuvo como objetivo identificar metabolitos asociados a la pérdida de masa muscular en pacientes con caquexia. El conjunto de datos estaba compuesto por 77 pacientes, distribuidos en 47 caquéxicos y 30 controles. Los IDs de las muestras incluían siglas como PIF, NETCR y NETL, que se consideraron como tipos de cáncer y se incorporaron al análisis para aportar una dimensión adicional al análisis posterior.

3.2. Construcción del objeto *SummarizedExperiment*

Se cargaron los datos de metabolómica desde un archivo .csv y se realizó una preprocesamiento inicial, que incluyó la creación de una columna adicional con el tipo de cáncer y la renombración de las muestras para distinguir claramente entre pacientes caquéxicos y controles en el ID. Posteriormente, se extrajeron los metadatos de las muestras y la matriz de concentración de metabolitos. Finalmente, se construyó el objeto de clase *SummarizedExperiment*, integrando la matriz de expresión y los metadatos de las muestras con la función *SummarizedExperiment()* siguiendo el tutorial de [CITA].

3.3. Normalización de los datos

Para estabilizar la varianza y reducir la influencia de valores extremos, las concentraciones de metabolitos fueron transformadas mediante logaritmo en base 2. Esta normalización se aplicó directamente a la matriz de expresión (concentración) contenida en el objeto *SummarizedExperiment* con la función log2().

3.4 Análisis univariante

Como primer análisis exploratorio, se calculó la mediana de las concentraciones normalizadas de metabolitos para cada muestra. . Posteriormente, se representaron las medianas mediante un boxplot por grupo, mostrando las diferencias en las distribuciones de las concentraciones globales de metabolitos entre ambos conjuntos de pacientes.

3.4. Análisis multivariante y Batch Effect

Se realizó un análisis de componentes principales (PCA) utilizando las concentraciones normalizadas de metabolitos. Las muestras fueron representadas diferenciando tanto el grupo de estudio (caquéxicos y controles) como el tipo de cáncer (PIF, NETCR, NETL) mediante codificación de colores, con el objetivo de explorar la variabilidad y evaluar posibles efectos de agrupación o batch effect. Además, se realizó un análisis de clustering jerárquico mediante distancias euclídeas, representando los resultados en un dendrograma con separación visual en dos grupos para ver si existe separación entre caquéxicos y controles.

3.5. Análisis de expresión diferencial

Se realizó un análisis de expresión diferencial para comparar las concentraciones de metabolitos entre pacientes caquéxicos y controles. Para cada metabolito, se aplicó un test t de Student, calculando además el log2 fold change entre ambos grupos. Los resultados se representaron mediante un volcano plot con la librería *EnhancedVolcano()*, resaltando aquellos metabolitos con cambios significativos en concentración.

4. Resultados

4.1 Caracterización inicial de los datos

Table 1: Metadatos de las muestras

	Muscle.loss	Cancer_Type
CA-1	cachexic	PIF
CA-2	cachexic	PIF
CA-3	cachexic	PIF

	Muscle.loss	Cancer_Type
CA-4	cachexic	NETL
CA-5	cachexic	PIF
CA-6	cachexic	PIF

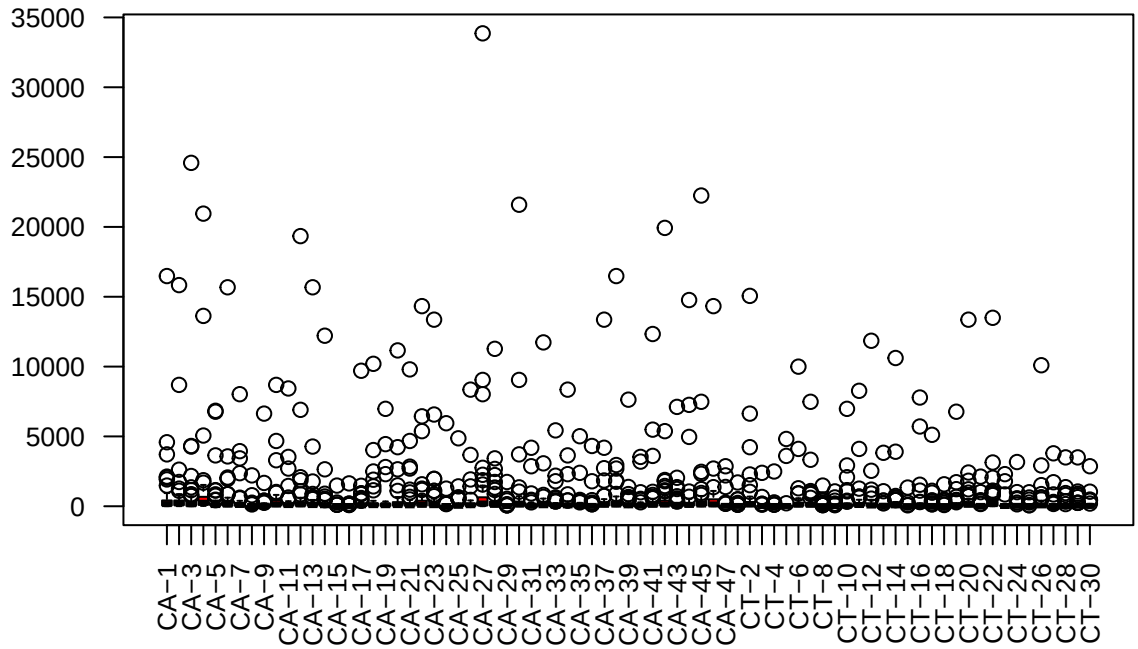
Table 2: Distribución de pacientes por grupo

Condición	Frecuencia
cachexic	47
control	30

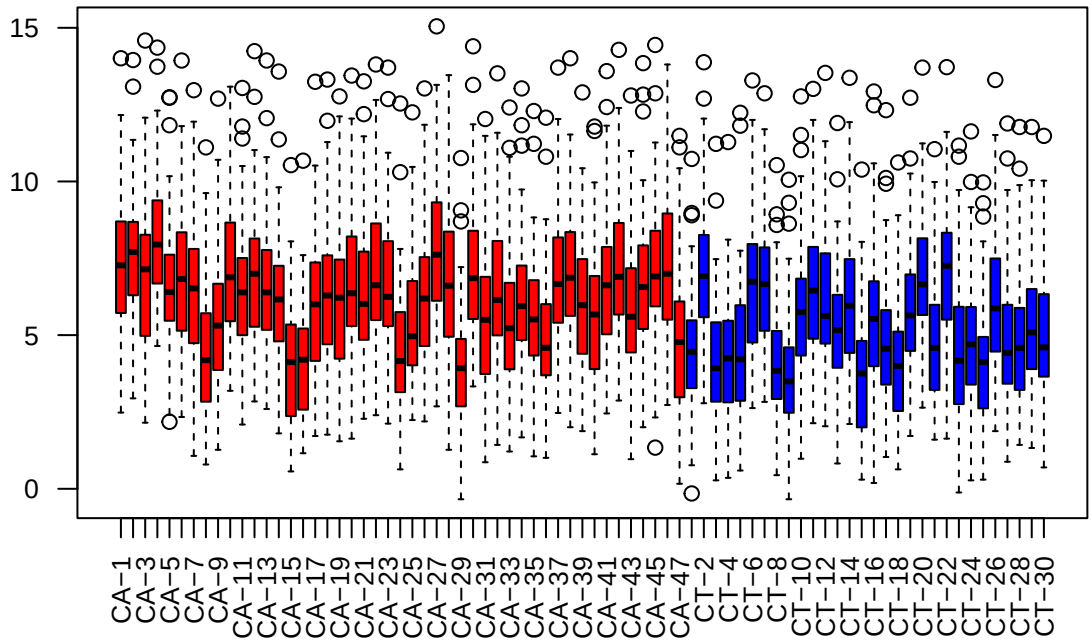
Table 3: Matriz de expresión

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

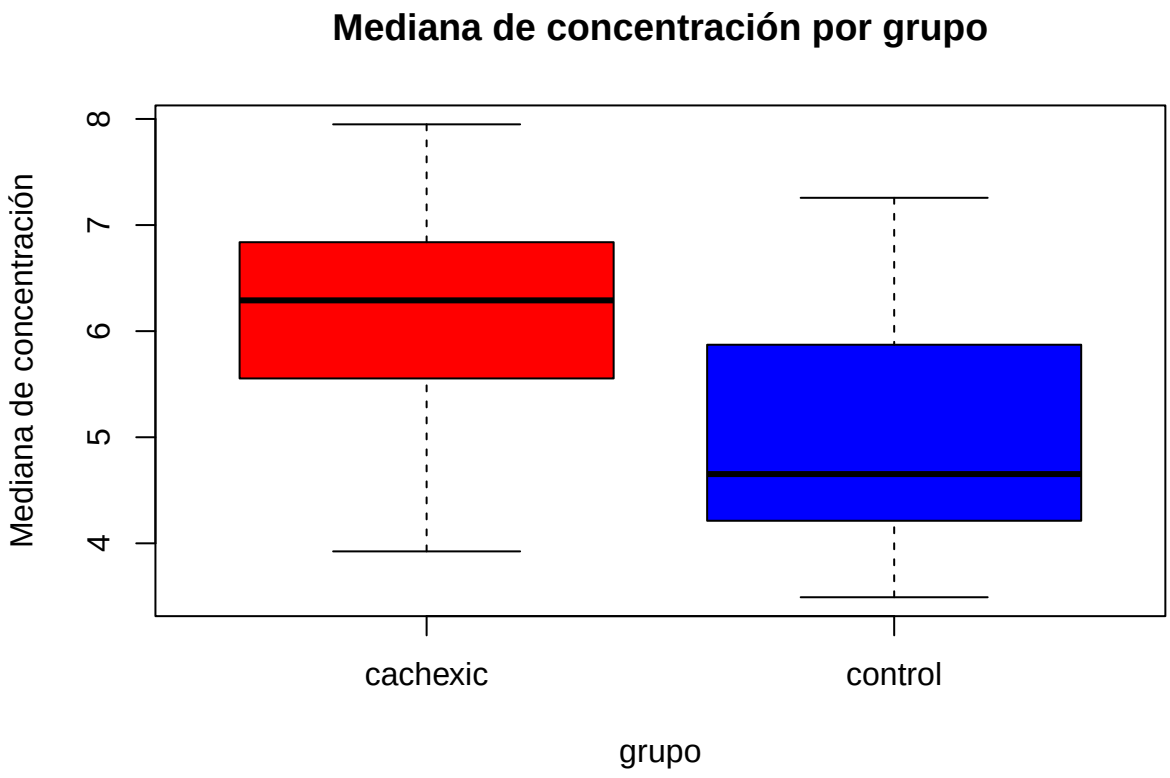
```
# Accedo a la matriz de expresión
expr_unn <- assay(se)

# Defino los grupos y sus colores
grupos <- colData(se)$`Muscle loss`  # por ejemplo
colores <- ifelse(grupos == "cachexic", "red", "blue")

# Creo el boxplot
boxplot(expr_unn,
        las = 2,
        col = colores,
        cex.axis = 0.8,
        main = "Distribución de los valores de concentración")
```




```r
#Normalización
expr_norm <- log2(assay(se))
# Crear boxplot
boxplot(expr_norm,
        las = 2,
        col = colores,
        cex.axis = 0.8,
        main = "Distribución de los valores de concentración")
```



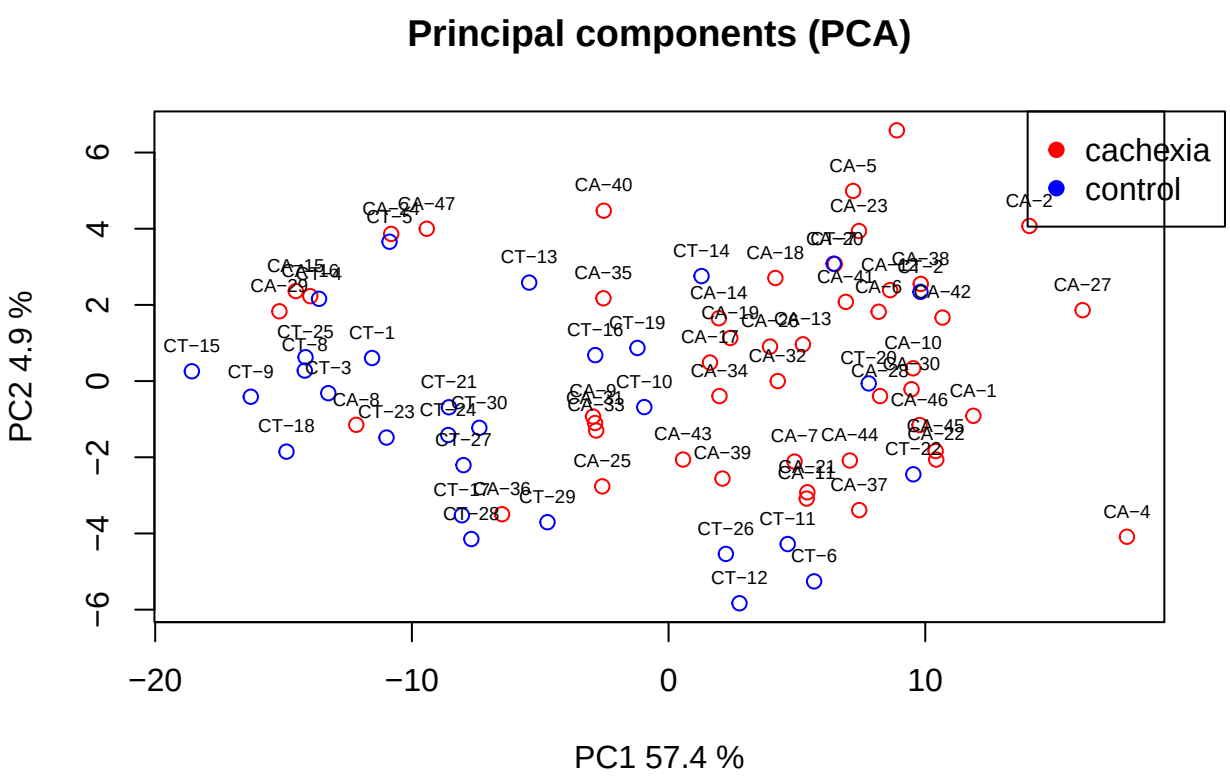

```r
# Obtener expresión y grupos
grupo <- colData(se)$`Muscle loss`
# Calcular medianas por muestra
medianas <- apply(expr_norm, 2, median)

# Crear boxplot por grupo
boxplot(medianas ~ grupo,
        col = c("red", "blue"),
        main = "Mediana de concentración por grupo",
        ylab = "Mediana de concentración")
```



PCA

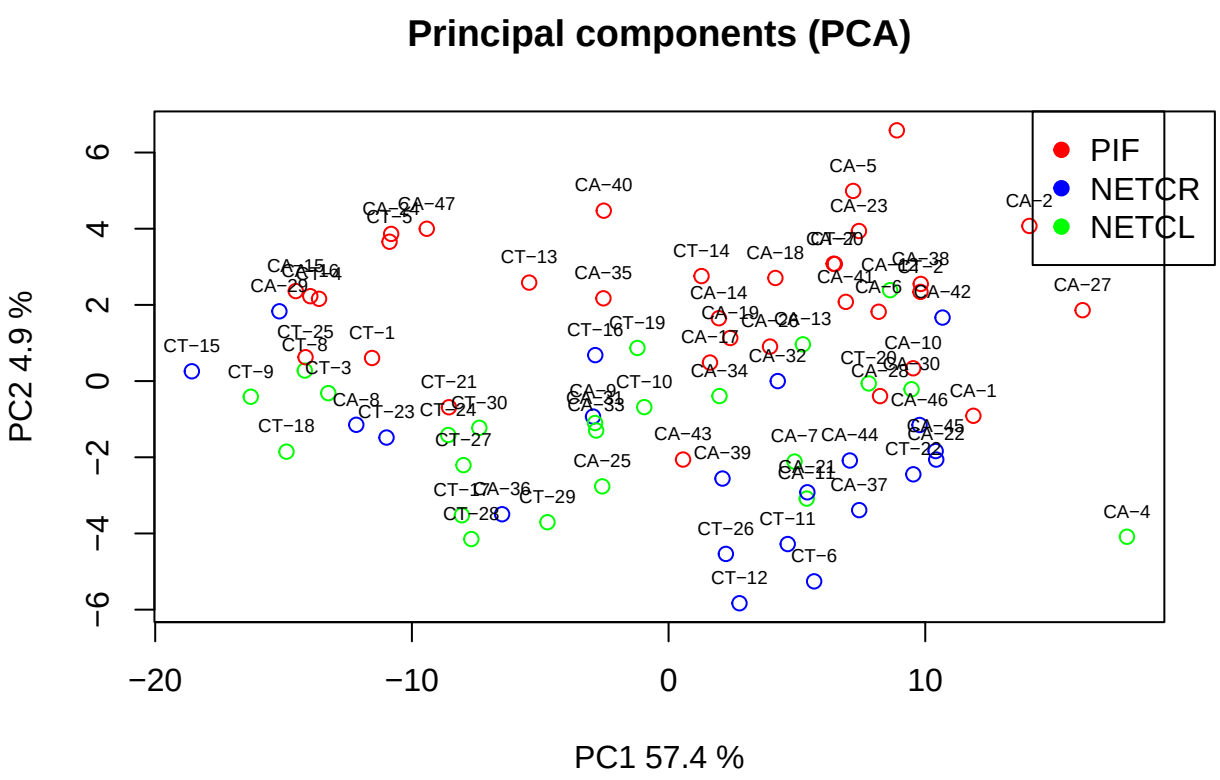
```
pc <-prcomp(t(expr_norm), scale=FALSE)
loads <- round(pc$sdev^2 / sum(pc$sdev^2) * 100, 1)
```

```
pacientes_ID <- cachexia[ ,1]

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

plot(pc$x[,1:2],xlab=xlab,ylab=ylab, col=colores,
     main ="Principal components (PCA)")

legend("topright", inset=c(-0.06,0),  # mueve la leyenda fuera
       legend=c("cachexia", "control"),
       col=c("red", "blue"),
       pch=19, xpd=TRUE)

text(pc$x[,1],pc$x[,2],pacientes_ID, pos=3, cex=.6)
```



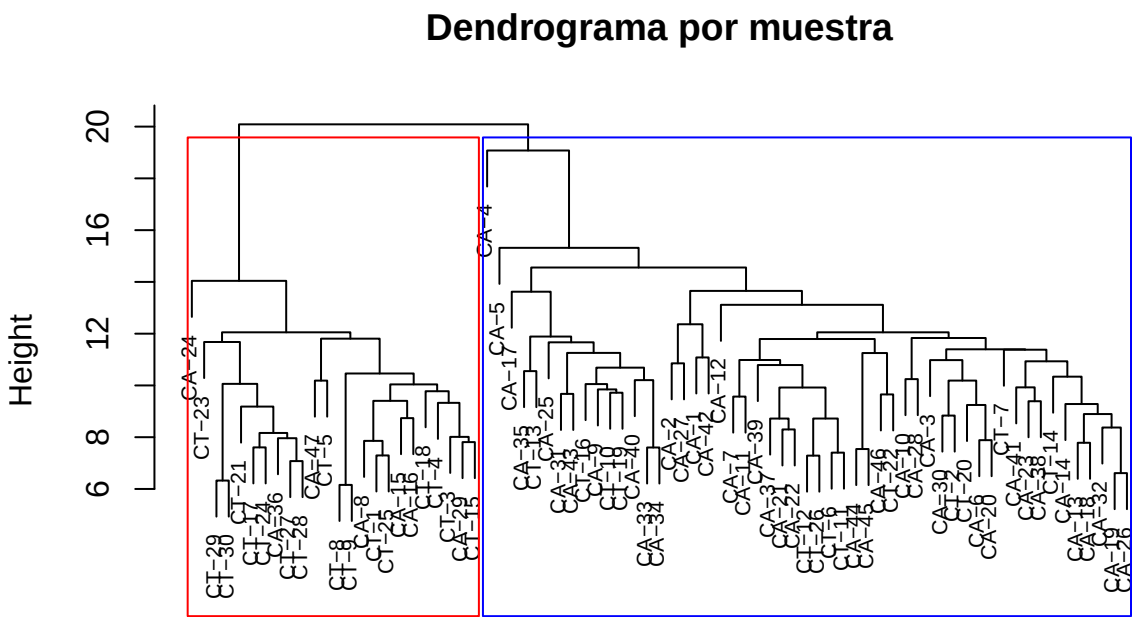
Batch Effect

```
grupo_batch <- colData(se)$Cancer_Type
colores_batch <- ifelse(grupo_batch == "PIF", "red",
                  ifelse(grupo_batch == "NETCR", "blue",
                         "green"))

plot(pc$x[,1:2],xlab=xlab,ylab=ylab, col=colores_batch,
     main ="Principal components (PCA)")

legend("topright", inset=c(-0.05,0),  # mueve la leyenda fuera
       legend=c("PIF", "NETCR", "NETCL"),
       col=c("red", "blue", "green"),
       pch=19, xpd=TRUE)

text(pc$x[,1],pc$x[,2],pacientes_ID, pos=3, cex=.6)
```



Dendograma

```
# Distancia y clustering jerárquico
d <- dist(t(expr_norm))
hc <- hclust(d, method = "average")

# Dibujo el dendrograma
hc$labels <- gsub("_.*", "", hc$labels)
plot(hc, main = "Dendrograma por muestra", cex = 0.7)
rect.hclust(hc, k = 2, border = c("red", "blue"))
```



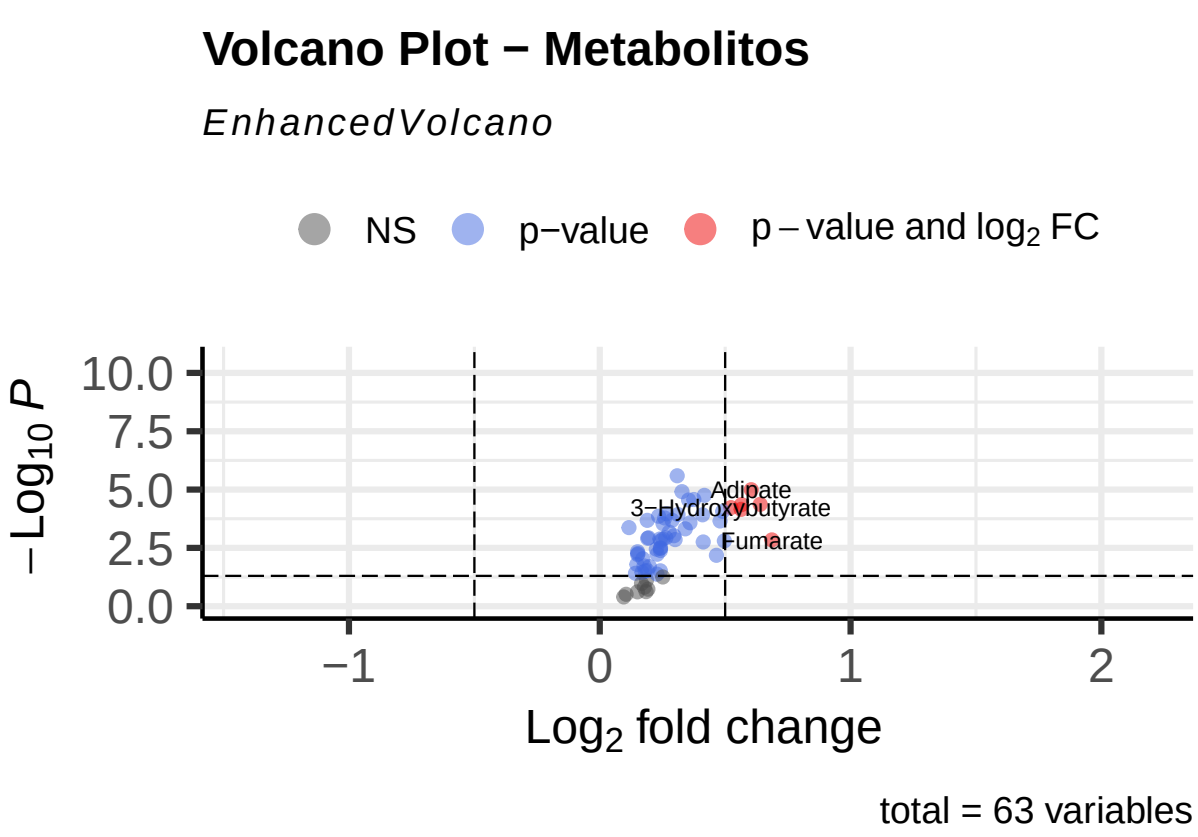
DEG

```
ttest <- function(x) {
  tt = t.test(x[1:47], x[48:77])
  return(c(
    t = tt$statistic,
    p.value = tt$p.value,
    log2FC = log2(mean(x[1:47]) / mean(x[48:77]))
  ))
}

ans <- apply(expr_norm, 1, ttest)
```

```
# Convertimos la matriz a data.frame
ttest_df <- as.data.frame(t(ans))

# Añadir nombres de los metabolitos (features)
ttest_df$feature <- rownames(ttest_df)
```

```
EnhancedVolcano(ttest_df,
                lab = ttest_df$feature,
                x = 'log2FC',
                y = 'p.value',
```

```
                pCutoff = 0.05,
                FCcutoff = 0.5,
                pointSize = 2,
                labSize = 3,
                title = 'Volcano Plot - Metabolitos')
```



https://rdrr.io/cran/specmine.datasets/man/cachexia.html