Anàlisis de dades òmiques: PEC1

Gabriel Regueira Huguet

2025-03-27

Contents

ABSTRACT

L'objectiu d'aquest estudi ha estat aprendre la funcionalitat de l'objecte *SummarizedExperiment* i familiaritzar-se amb la ruta d'un anàlisi exploratori de dades metabolòmiques. Per fer-ho, s'ha utilitzat un dataset provinent d'un estudi metabolòmic que analitzava perfils metabolòmics de diferents concentracions de metabòlits concentrats en la orina de pacients amb *Cachexia* o pacients control. Aquest dataset amb 77 pacients i 63 metabòlits mesurats s'ha reorganitzat mitjançant l'objecte de classe *SummarizedExperiment* per a fer-ne posteriorment un anàlisi exploratori. L'anàlisi exploratori ha inclòs anàlisis univariants (Boxplots, Boxplots múltiples) i anàlisis multivariants (PCA, Clustering jeràrquic i Heatmap), utilitzant eines no paramètriques perquè les dades no seguien una distribució normal (tests de Wilcoxon). Els resultats han mostrat diferències estadísticament significatives en les concentracions de la majoria de metabòlits entre ambdós grups, suggerint que aquest perfil metabolòmic podria tenir un paper important en la identificació de pacients amb *cachexia*.

OBJECTIUS

Els objectius d'aquest informe són familitaritzar-se amb la ruta d'un anàlisis de dades metabolòmiques, desde la seva importació de dades crues fins al posterior anàlisis exploratori estadístic i interpretació dels resultats. Per a això, serà necessari un seguit de sub-objectius:

- Familiaritzar-se amb l'OOP S4 *SummarizedExperiment* (Bioconductor, 2024) i crear-ne un amb les dades que s'han triat.
- Aprendre a utilitzar el repositori github per fer control de versions git, acabant amb un repositori de Github contenint els materials necessaris per a clonar tota la informació de l'estudi.
- Indagar en les dades per trobar patrons mitjançant el llenguatge de programació R i familiaritzar-se amb els preprocessats de dades necessaris i mètodes que s'utilitzen per fer un anàlisis estadístic exploratori de dades metabolòmiques.

MÈTODES

Per a fer aquesta PEC1 s'han utilitzat un conjunt de dades provinents d'un estudi sobre pacients amb càncer, que inclou concentracions de 63 metabòlits diferents en mostres d'orina expressades en μM. i informació clínica sobre pacients amb *cachexia* i pacients control. Les dades es van importar en format *.csv* des del repositori de GitHub proporcionat per l'enunciat de la PEC1. Mitjançant el programa RStudio, es van importar les dades a través de l'url del mateix repositori i es va estructurar l'informació en un objecte de classe *SummarizedExperiment*, integrant la matriu de dades quantitatives i les metadades clíniques de cada pacient. L'anàlisis exploratori estadístic va incloure un estudi univariant (resums numèrics, boxplots i proves t) i un estudi multivariant mitjançant un anàlisis de components principals (PCA). Seguint amb la exploració de dades, es va realitzar un clustering jeràrquic i un heatmap per identificar patrons de similitud entre les mostres. Tots els anàlisis es van realitzar mitjançant el programa RStudio i el llenguatge de programació d'R.

RESULTATS

1- Seleccionem un dataset de metabolòmica:

La *cachexia* és un síndrome metabòlic complex, que es mostra habitual en pacients amb càncer. Aquest síndrome es caracteritza per una pèrdua de massa muscular i/o greixosa, inflamació sistèmica, alteracions hormonals i en el metabolisme enegètic. Aquesta pèrdua de massa muscular es tradueix a un catabolisme muscular accelerat, que proboca l'alliberament de aminoàcids com la valina, leucina, alanina, etc. Aquest síndrome proboca que el pacient es trobi en un estat de semi-fam metabòlica, que proboca que hi hagi una demanda energètica elevada i alguns metabòlits intermedis del metabolisme energètic s'acumulen (3-hydrpxybutyrate, pyroglutamate, glutamine).

```
#Importem directament les dades crues (raw) desde el repositori de github que proporciona l'enunciat:
url_github <- "https://raw.githubusercontent.com/nutrimetabolomics/metaboData/refs/heads/main/Datasets/
human_cachexia_data <- read.csv(url_github, header = TRUE, sep = ",") #Separació per comes tal i com es
```

Observem un dataset on les files són les mostres (pacients) i les columnes són els metabòlits analitzats. També podem observar que hi ha una variable que separa els pacients en dos grups, pacients amb *cachexia* i pacients control.

2- Crear un objecte *SumarizedExperiment*:

Per a crear un objecte *SummarizedExperiment* necessito una matriu de dades quantitatives (amb les concentracions de metabòlits per pacient) i un *colData* amb informació de les mostres, que en aquest cas és el tipus de grup que pertany cada pacient (*Muscle.loss: cachexia/control*).

```
#Matriu de dades numèriques
assay_data1_ <- human_cachexia_data[, -(1:2)] #Eliminem les dues primeres columnes (mostres i grup)
rownames(assay_data1_) <- human_cachexia_data$Patient.ID #Assignem com a nom de fila els identificadors
assay_data1 <- as.matrix(assay_data1_)
View(assay_data1) #Observem que tenim el nom de les files assignats als identificadors dels pacients (P
```

```
#colData: Informació sobre les mostres (pacients): Muscle.loss
col_data1 <- data.frame(
  Patient.ID = human_cachexia_data$Patient.ID, #Agafem els pacients,
  Muscle.loss = human_cachexia_data$Muscle.loss #I el grup al que pertanyen
)
rownames(col_data1) <- human_cachexia_data$Patient.ID #Assignem coma nom de fila els indentificadors co
View(col_data1)
```

Abans de fer el *SummarizedExperiment*, necessitem tenir una matriu numèrica on les mostres estiguin com a columnes en comptes de com a files, de la mateixa manera els metabòlits com a files en comptes de com a columnes:

```
#Trasposem la matriu per tenir metabolits (files) x pacients (columnes)
assay_data1t <- t(assay_data1)
View(assay_data1t) #Automàticament ja cambia rownames per colnames quan trasposem la matriu
```

Ara ja tenim la matriu de dades numèriques llesta per a fer *SummarizedExperiment*

```
library(SummarizedExperiment)
```

```
cachexia_se <- SummarizedExperiment(
  assays = list(counts = assay_data1t), #Matriu de dades
  colData = col_data1 #Grup per pacient
)
cachexia_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(77): PIF_178 PIF_087 ... NETL_003_V1 NETL_003_V2
## colData names(2): Patient.ID Muscle.loss
```

Una vegada creat el *SummarizedExperiment*, el guardarem en un arxiu en format .Rda com indica l'enunciat:

```r
save(cachexia_se, file = "cachexia_se.rda") #Guardem l'arxiu al repositori
```

Diferències *ExpressionSet* i *SummarizedExperiment*:

ExpressionSet ha estat durant molt temps el format clàssic per analitzar dades de miacroarrays, només admet una única matriu de dades (*exprs*). És molt útil, però està pensat per un tipus específic de dades i no té tanta flexibilitat. En canvi l'objecte *SummarizedExperiment* és més potent, ja que és capaç de gestionar més tipus de dades (comptes, intensitats, etc) i pot contenir múltiples matrius (en *assays*) i és compatible amb dades més complexes, és l'OOP estàndard actual per a estudis RNa-seq, proteòmica i metabolòmica. Tots dos objectes són molt útils per organitzar les dades de manera integrada i sincronitzada, però *SummarizedExperiment* ho fa amb més flexibilitat i amb una estructura més moderna.

3- Anàlisis exploratori:

```r
#S'ha fet un resum estadístic per a cada metabòlit (mínim, mitjana, màxim, etc), posem els 5 primer com
apply(assay(cachexia_se)[1:5, ], 1, summary) #Resum numèric dels metabòlits (5 primers)
dim(cachexia_se)
```

```r
#Comprovem que no hi hagi valors faltants (NA) en la matriu de dades
anyNA(assay(cachexia_se))
```

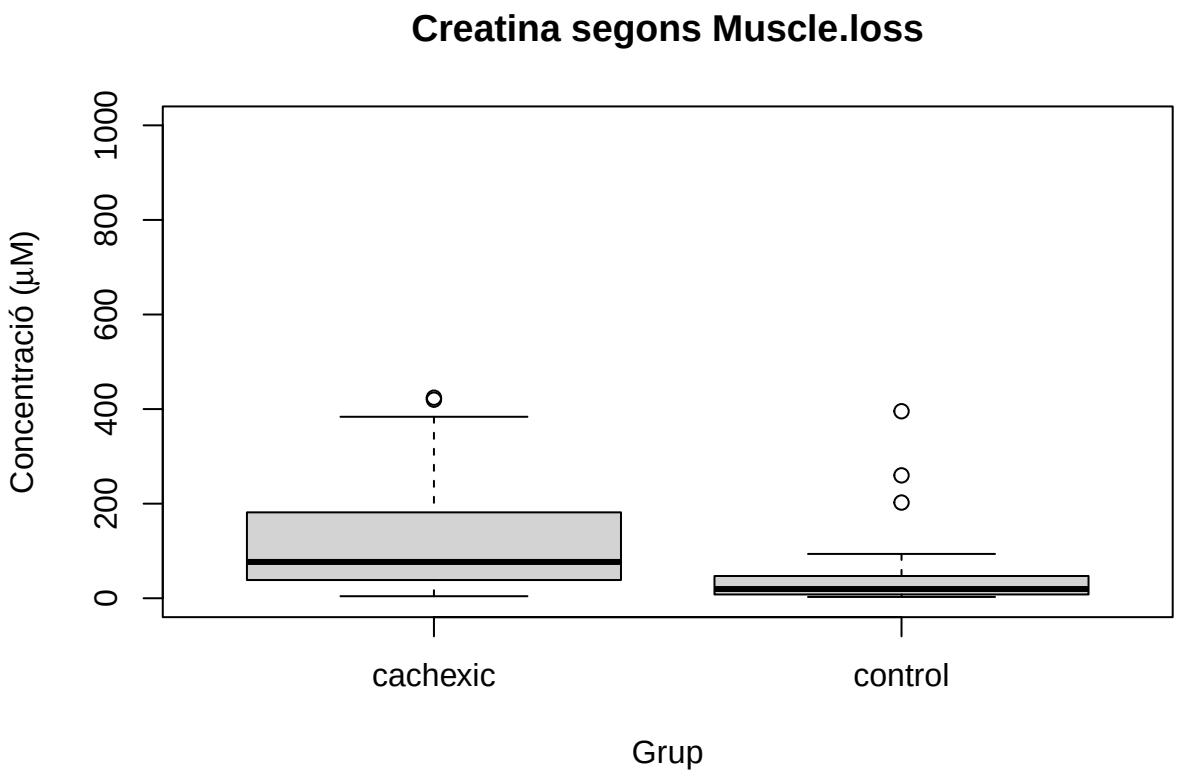
```
## [1] FALSE
```

Podem observar de forma general que aquestes dades consten d'un OOP *SummarizedExperiment* on la matriu de dades està format per 77 pacients (columnes) els quals estàn dividits pel grup "Muscle.Loss" i 63 metabòlits (files) que són les concentracions de diferents metabòlits analitzades en les mostres d'orina dels pacients.

Anàlisis univariant: Creatina

Hem realitzat apart un anàlisi del test *Shapiro-Wilk* per saber si les variables numèriques (metabòlits) segueixen una distribució normal. Resulta que cap d'elles segueix una distribució normal i, per tant, haurem de procedir amb l'anàlisis sense asumir normalitat.

```r
creatina <- assay(cachexia_se)["Creatine", ] #Extraiem les concentracions del metabòlit creatina
muscle_loss <- colData(cachexia_se)$Muscle.loss #Extraiem el grup Muscle.loss
boxplot(creatina ~ muscle_loss,
        main = "Creatina segons Muscle.loss",
        xlab = "Grup",
        ylab = expression("Concentració ("*mu*"M)"),
        ylim = c(0, 1000))
```



Fem un test *Wilcoxon* per veure si les distribucions de creatina són diferents entre grups (*Muscle.loss*), sense assumir normalitat:

```
wilcox.test(creatina ~ muscle_loss)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  creatina by muscle_loss
## W = 1077, p-value = 0.0001042
## alternative hypothesis: true location shift is not equal to 0
```

Mitjançant aquest anàlisi bàsic podem observar que el metabòlit creatina mostra una diferència significativa en la concentració entre els grups *Muscle.loss*. Observem que la mitjana en el grup que tenen *cachexia* (pèrdua constant de massa muscular) és significativament superior a la del grup control. Aquests resultats poden indicar que la concentració de creatina en la orina podria estar relacionada amb l'estat de cachexia i, per tant, podria ser un potencial marcador per ajudar a diagnosticar aquesta malaltia.

Boxplot múltiple: metabòlits més rellevants

Seguidament, seguirem amb l'anàlisis estadístic descriptiu mitjançant un boxplot múltiple. Com que no podem fer un boxplot dels 63 metabòlits, farem un test de *Wilcoxon* univariant per a cada metabòlit i seleccionarem els 4 metabòlits que tinguin p-valors més baixos (més significació).

```r
metabolits <- assay(cachexia_se) #Assignem metabolits
group <- colData(cachexia_se)$Muscle.loss #Assignem group a la variables Muscle.loss
#Fem un test Wilcoxon per cada metabòlit i guardem els p-valors dels tests
p_valors <- apply(metabolits, 1, function(x) {
  suppressWarnings(
  tryCatch(wilcox.test(x ~ group)$p.value, error = function(e) NA) #Agafem els p_valors dels tests de c
  )
})
#Ordenem els metabòlits segons els p-valors que hagin donat els tests
p_valors_ordenats <- sort(p_valors)
top_metabolits <- names(p_valors_ordenats) [1:4]
top_metabolits
```

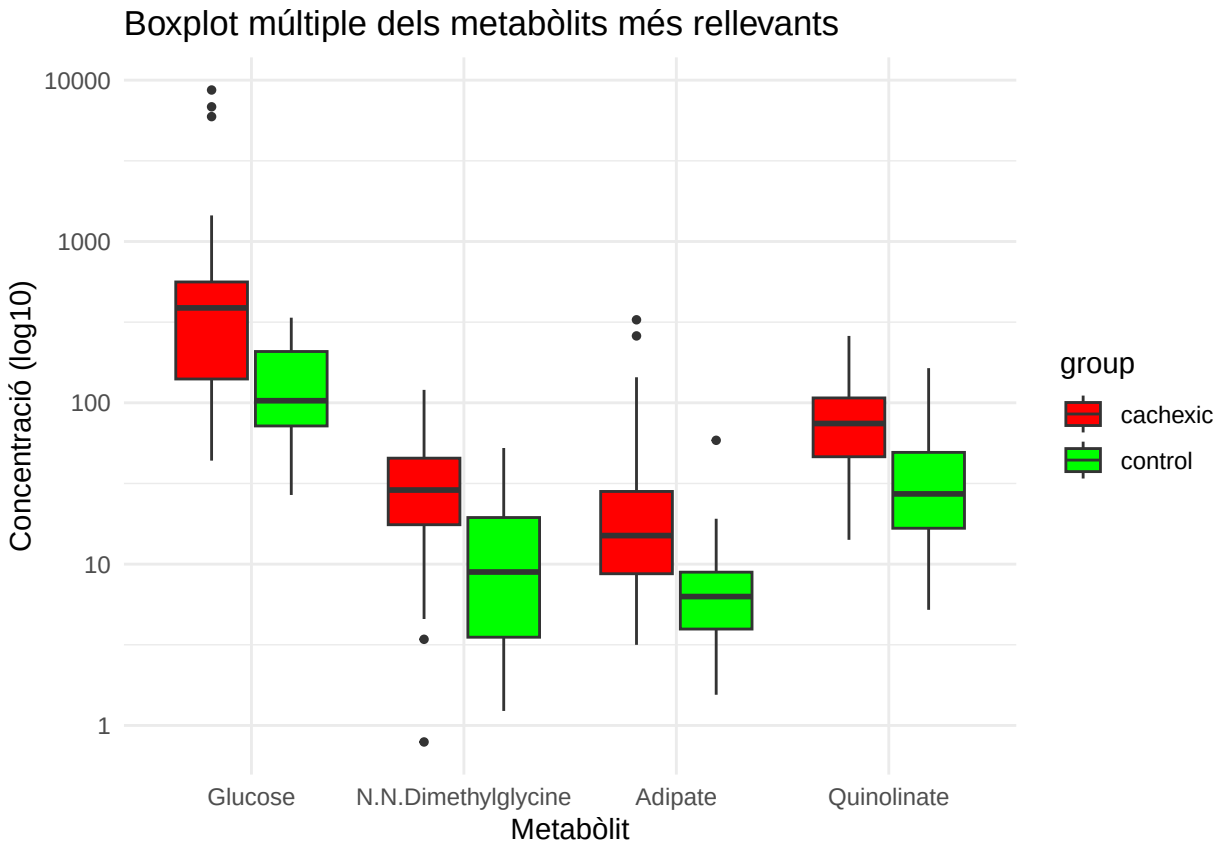
```
## [1] "Quinolinate"         "Glucose"             "Adipate"
## [4] "N.N.Dimethylglycine"
```

```r
sum(p_valors < 0.05, na.rm = TRUE) #coompta quants són significatius
```

```
## [1] 55
```

Aquests són els metabòlits que han donat més nivell de significació fent el test no paramètric de *Wilcoxon* segons la variable grup *Muscle.loss*. Per tant, haurien de ser els que tenen diferències més significatives de concentracions segons si els pacients tenen *cachexia* o no. També cal recalcar que, **dels 63 metabòlits de l'estudi, 55 metabòlits han donat diferències significatives amb el test de *Wilcoxon*** . Donat que hi han concentracions motl diferents entre els metabòlits, apliquem una escala logarítmica per tal que millor la visualització del *Boxplot múltiple*.

```r
library(reshape2)
library(ggplot2)
#Seleccionem els metabòlits que ens han sortit
top_metabolits <- c("Glucose", "N.N.Dimethylglycine", "Adipate", "Quinolinate")
#Extraiem la matriu només amb els metabòlits seleccionats
top_data <- assay(cachexia_se)[top_metabolits, ]
#Preparem les dades per fer ggplot2 (passem les mostres a les files en comptes de les columnes i anyadi
top_data_prep <- as.data.frame(t(top_data))
top_data_prep$group <- colData(cachexia_se)$Muscle.loss #Anyadim la columna grup
#Format compatible amb bloxplot
data_metabolits <- reshape2::melt(top_data_prep, id.vars = "group",
                                  variable.name = "metabòlit",
                                  value.name = "concentració")
#Amb les dades preparades, procedim a fer el boxploit múltiple
ggplot(data_metabolits, aes( x= metabòlit, y = concentració, fill = group)) +
  geom_boxplot(outlier.size = 1) +
  labs(title = "Boxplot múltiple dels metabòlits més rellevants",
       x = "Metabòlit",
       y = "Concentració (log10)") +
  scale_fill_manual(values = c("cachexic" = "red", control = "green")) + #Separem grups per color
  scale_y_log10() + #Apliquem escala logaritmica per fer comparables metabolits amb concentracions mass
  theme_minimal()
```



Aquest gràfic mostra les diferències de concentracions dels 4 metabòlits que presenten més diferències significatives segons la variable categórica *Muscle.loss*. Tal i com s'observa en el gràfic, els 4 metabòlits tenen majors concentracions en els individus que presenten la malaltia *cachexia* que en els individus del grup control.

Anàlisis de Components Principals (PCA)

Mitjançant aquest tipus d'anàlisis, l'objectiu serà reduïr la dimensió de les dades i visualitzar si les mostres s'agrupen segons "Muscle.Loss" (*cachexia/control*) basant-se en els seus perfils metabolòmics. Tenint en compte que les concentracions de metabòlits varien molt, és recomanable estandaritzar (escalar) les dades abans de fer la PCA, ja que sinó les variables amb majors rangs tindràn més pes.

```
#Primerament, trasposem la matriu de l'objecte per tenir les mostres com a files i els metabòlits com a
t_data <- t(assay(cachexia_se))
#És recomanable escalar les variables quan estàn a diferents escales , en el nostre cas algun metabòlit
pca_resultats <- prcomp(t_data, scale. = TRUE)
summary(pca_resultats)$importance[, 1:5] #Mostrem els 5 components principals més importants (dels 63PC
```

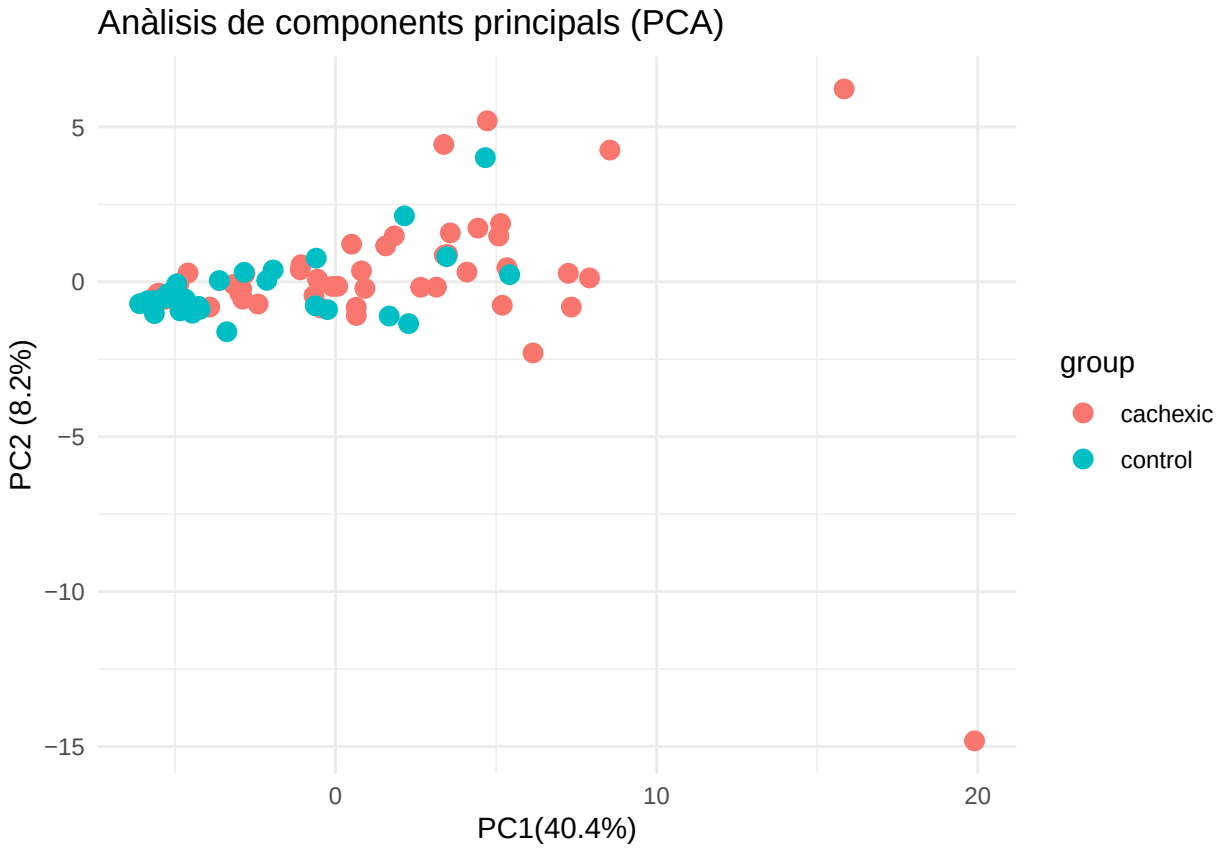
```
##                            PC1      PC2      PC3      PC4      PC5
## Standard deviation     5.04667 2.270128 1.833107 1.747276 1.659056
## Proportion of Variance 0.40427 0.081800 0.053340 0.048460 0.043690
## Cumulative Proportion  0.40427 0.486070 0.539410 0.587870 0.631560
```

```
var_explicada <- summary(pca_resultats)$importance[2, ] * 100 #Seleccionem PC1 i PC2
```

Observem en els resultats de l'anàlisis de components principals que els dos primers ja tenen una variabilitat

del **48.61%**, que ja es considera bastant alta. Decidim quedar-nos amb els dos primers components principals i utilitzarem aquests per a obtenir una representació de les dades en una dimensió reduïda:

```
cach_control <- colData(cachexia_se)$Muscle.loss
pca_d <- as.data.frame(pca_resultats$x) #Cada fila representa una mostra i cada columna un PCA
pca_d$group <- cach_control #Afegim la classe de cada mostra
library(ggplot2)
ggplot(pca_d, aes(x = PC1, y = PC2, color = group)) + #Separem per grup segons el color
geom_point(size = 3) +
labs(
title = "Anàlisis de components principals (PCA)",
x = paste0("PC1(", round(var_explicada[1], 1), "%)"),
y = paste0("PC2 (", round(var_explicada[2], 1), "%)")
) + theme_minimal()
```

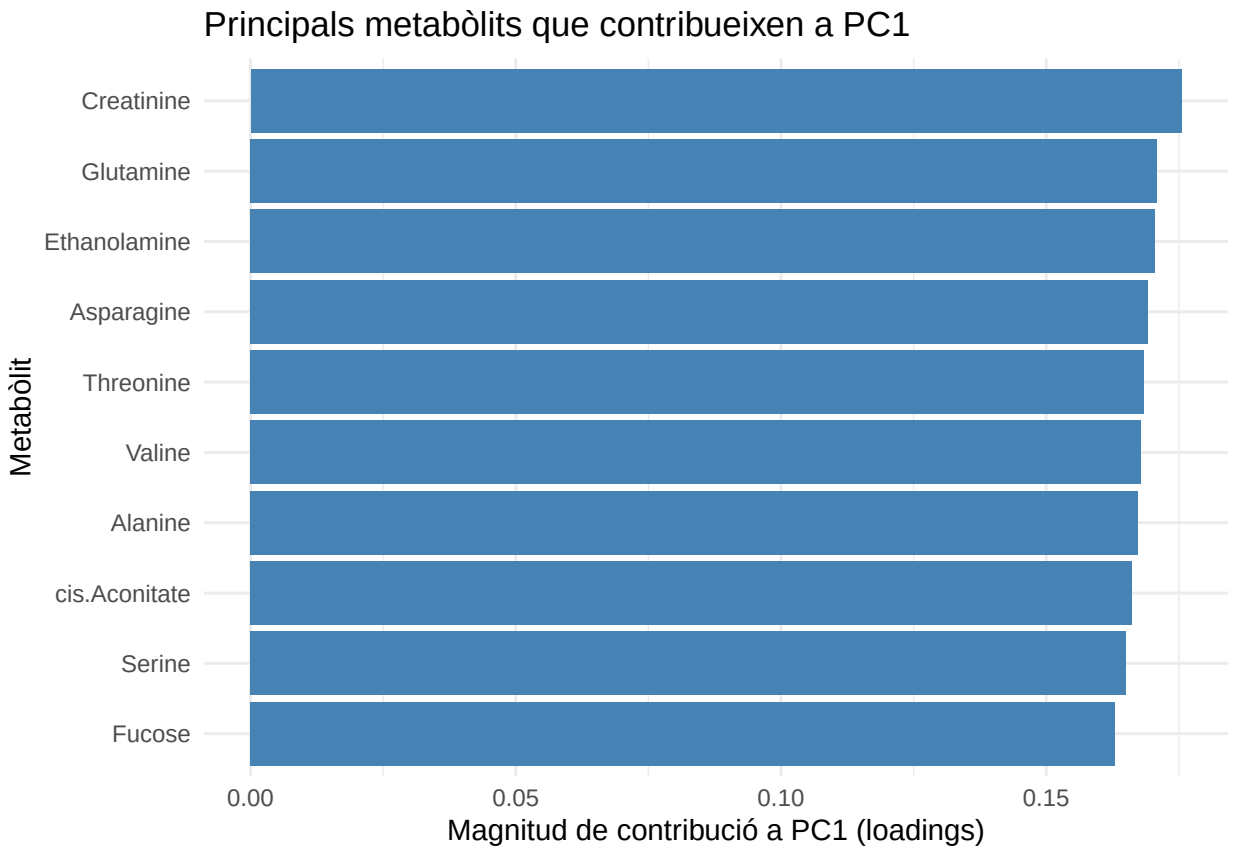


S'ha realitzar un anàlisi de components principals sobre la matriu de concentracions de metabòlits. Prèviament s'han centrat i escalat les dades per a evitar que les diferències d'escala entre les variables afectin l'anàlisi. Els dos primers components principals, com es pot observar, expliquen gairebé un 50% de la variança total (48.6%). Observem en el gràfic que les mostres del grup *cachexic* (vermells) tendeixen a situar-se en valors positius de PC1, mentre que les mostres *control* (blaus) es concentren en valors al voltant de zero. Això pot indicar que la variabilitat capturada per PC1 està relacionada amb les diferències entre els dos grups de *Muscle.loss*, tot i que hi ha molta superposició entre les mostres i no es pot veure una diferència clara.

La magnitud de la contribució de cada variable a les PC són els seus "loadings" en cada PC. Els autovectors (eigenvectors) associats a la matriu de covariança són els loadings, indiquen quina direcció prenen els nous components i quines variables (metabòlits) contribueixen més.

```r
#creem un data frame amb els "loadings" (magnitud de contribució de cada metabòlit al component princip
pca_resultats <- prcomp(t_data, scale. = TRUE)
loadings_pca <- as.data.frame(pca_resultats$rotation) #assignem els loadings
loadings_pca$metabolit <- rownames(loadings_pca)
```

```r
top_PC1 <- loadings_pca[order(abs(loadings_pca$PC1), decreasing =TRUE), ][1:10, ] #Agafem els 10 metabò
#Grafic
ggplot(top_PC1, aes(x = reorder(metabolit, PC1), y = PC1)) +
  geom_col(fill = "steelblue") +
  coord_flip() + #Cambiem els eixos per a millor visualització
  labs(title = "Principals metabòlits que contribueixen a PC1",
       x = "Metabòlit",
       y = "Magnitud de contribució a PC1 (loadings)") +
  theme_minimal()
```



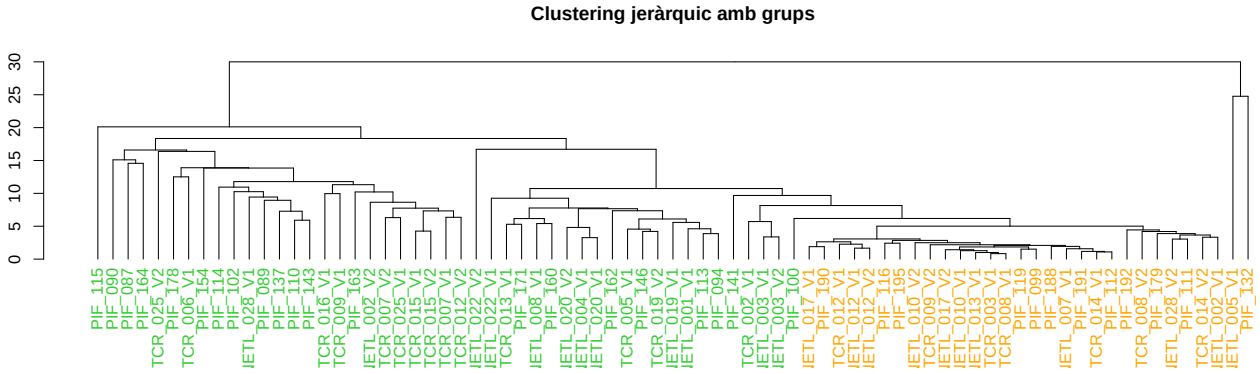
Aquest gràfic mostra els 10 metabòlits que més contribueixen a la variança capturada pel primer component principal (PC1). Com hem pogut observar prèviament, el PC1 és el component principal del qual la seva direcció recull la major part de la variabilitat de les dades, i els valors *loading* indicarien quina força té cada metabòlit en definir aquesta direcció. Podem oobservar com la majoria de metabòlits contribueixen de forma gairebé equitativa a la PC1, destaquem la *Creatine*, que és la que contribueix més. Això té coherència amb l'anàlisi anterior, on ja havíem vist que aquest metabòlit mostrava diferències significatives segons el grup (*cachexia/control*).

Clustering jeràrquic

El clustering jeràquic és un potent recurs per a l'anàlisis exploratori de dades, porporcionant mètodes potents i flexibles per descubrir grups en les dades. Com s'ha mencionat anteriorment, els resultats del test no paramètric de *Wilcoxon* mostren que 55 dels 63 metabòlits de l'estudi presenten diferències significatives. Això suggereix que gairebé totes les variables de l¡estudi contenen informació rellevant per a la classificació, per tant, es va optar per no filtrar i incloure tots els metabòlits a l'anàlisis de clustering jeràrquic.

```
library(dendextend)
```

```
dades_s <- scale(t_data) #Escalem la matriu abans de clusteritzar (mitjana = 0, sd = 1)
dist_mostres <- dist(dades_s, method = "euclidean") #Calculqem la matriu de distàncies (eucledian)
hc <- hclust(dist_mostres, method = "complete") #Mètode de distància escollit: "Complete link", màxim d
grups <- colData(cachexia_se)$Muscle.loss #grups per etiquetal segons Muscle.loss
#Gràfic del dendograma
dend <- as.dendrogram(hc)
labels_colors(dend) <- ifelse(grups == "cachexic", "limegreen", "orange") #Dividim les mostres segons e
plot(dend,
     main = "Clustering jeràrquic amb grups",
     cex = 0.8)
```



Observem en el dendrograma que hi ha una separació visible entre els dos grups principals (*cachexia* i *control*), tot i que no està perfectament separada perquè algunes mostres *cachexia* (verd) queden barrejades amb *control* (taronja). Això pot indicar que pot haver efectes tècnics que desconeixem i/o que només alguns metabòlits separen clarament els dos grups (no tots).

Heatmap

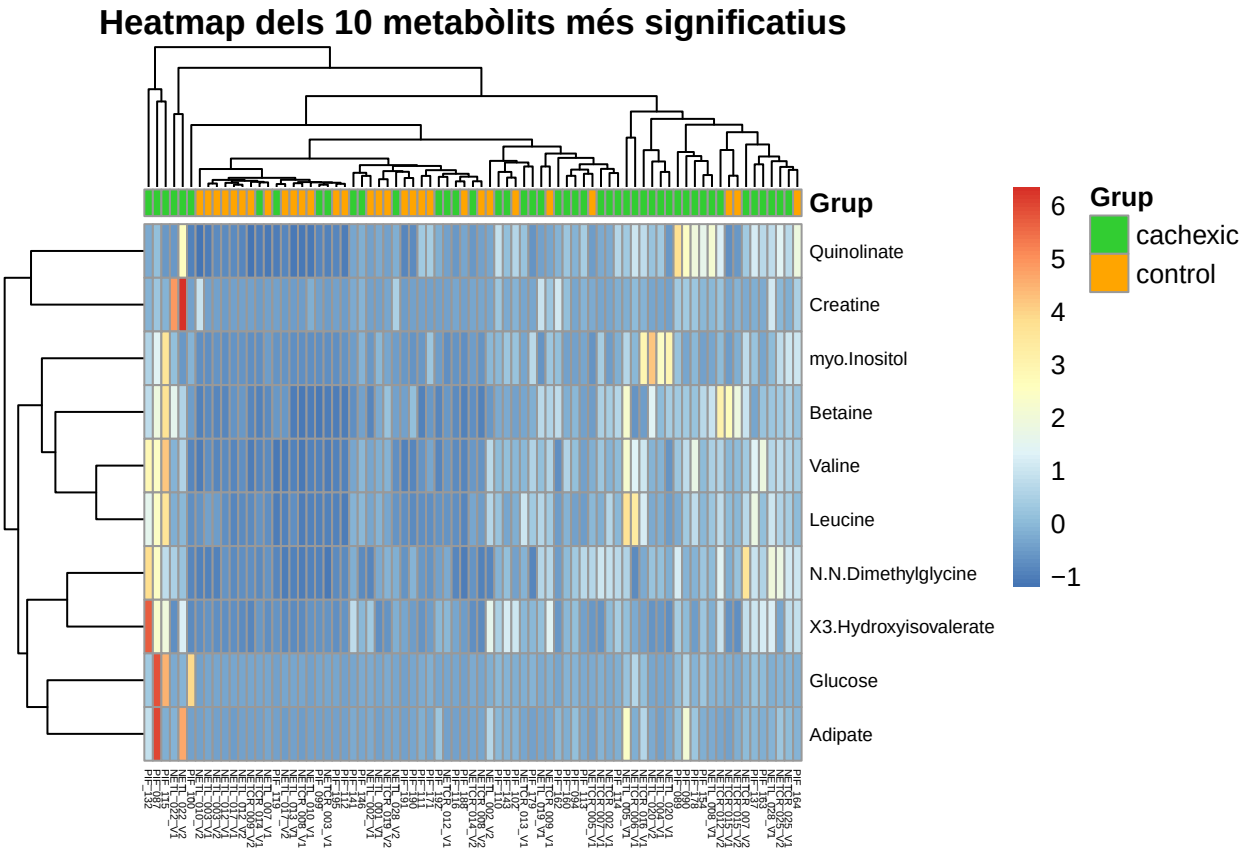
Un heatmap amb 63 variables (meatbòlits) seria massa sorollós i difícil d'interpretar. Per tant, abans de fer el heatmap farem una selecció prèvia dels 10 metabòlits (variables) més significatius.

```
#Com ho hem fet anteriorment, ja tenim els p-valors ordenats dels metabòlits
p_valors_ordenats <- sort(p_valors)
top_metabolits_heatmap <- names(p_valors_ordenats)[1:10] #Seleccionem els 10 metabòlits més significati
#extraim les dades només per als meta`+bolits més significatius
mat <- metabolits[top_metabolits_heatmap, ] #10 files (metabòlits) x 77 mostres (pacients)
#Escalem pels metabòlits (mitjana 0, desviació 1)
mat_scaled <- t(scale(t(mat))) #trasposem, escalem i tornem a transposar després
```

El heatmap mostratà les mostres (pacients) i els metabòlits, però no sap quin grup pertany cada mostra. Per tant, hem de fer que el mapa pugui caracteritzar les mostres segons el grup que pertany, li hem de donar la informació.

```r
anotacions <- data.frame(Grup = grups) #Creem un petit dataframe amb la columna grup (cachexia/control)
rownames(anotacions) <- colnames(mat_scaled) #Així el heatmap sabrà que x columna és la mostra PIC_XXX
library(pheatmap)
```

```r
pheatmap(mat_scaled, #Matriu de dades de 10 meatbòlits per 77 mostres (pacients)
         annotation_col = anotacions, #Afegeix una linea de colors a dalt del mapa indicant si la mostr
         annotation_colors = list(
           Grup = c(cachexic = "limegreen", control  ="orange") #Definim el color per cada grup
         ),
         scale = "none", #None, ja hem escalat els valors manualment
         clustering_distance_rows = "euclidean", #Mètode per agrupar metabòlits
         clustering_distance_cols = "euclidean", #Mètode per agrupar mostres
         clustering_method = "complete", #clustering jeràrquic
         main = "Heatmap dels 10 metabòlits més significatius",
         fontsize_row = 7, #Tamany text dels metabòlits significatius
         fontsize_col = 4) #Tamany text de les mostres
```



Com que hem escalat la matriu de dades dels metabòlits, per cada fila de metabòlit la mitjana és 0 i la desviació estàndard és 1. D'aquesta manera la majoria de valors d'un metabòlit queden aprop del 0, però si hi ha algun valor molt alt comparat amb la mitjana, aquesta destacarà sobre la resta i mostrarà una coloració més llunyana del blau/blanc i s'aproparà al vermell. D'aquesta manera, amb el heatmap podem veure quins valors de metabòlits destaquen sobre la resta.

El dendrograma de dalt mostra com les mostres (pacients) s'agrupen segons la semblança dels seus perfils metabolòmics, d'aquesta manera veiem que les mostres de color verd que pertany al grup que té *cachexia* tendeixen a agrupar-se a la dreta, on es mostren valors dels metabòlits més elevats que les seves mitjanes (colors allunyats del blau). Mentre que les mostres de taronja que pertanyen als pacients control, tendeixen a

agrupar-se a l'esquerra, amb perfils metabolòmics més propers a la mitjana (color blau). Tot i així, observem en l'extrem esquerre que s'acumulen 5 mostres *cachexia* amb uns metabòlits molt per sobre de la mitjana.

Com ja havíem vist en els anàlisis anteriors, aquest patró reforça la idea que els pacients amb *cachexia* semblen presentar perfils metabolòmics diferenciats, amb concentracions més elevades en diversos metabòlits rellevants. Si ens fixem, de forma general la majoria de mostres del grup control presenten metabòlits amb concentracions tirant més a la normalitat (color blau), mentre que les mostres *cachexia* observem que els seus metabòlits ja tenen alteracions en la coloració mostrant mitjanes de concentracions més elevades.

DISCUSSIÓ

Els resultats observats al heatmap i la resta d'anàlisis són consistents amb la literatura científica sobre el síndroma *cachexia*. Ja que *cachexia* és un síndrome caracteritzat per una gran desregulació metabòlica, un augment de la degradació de proteïnes musculars i una activació de la gluconeogènesi i alteració de les vies energètiques (Evans et al., 2009; Argilés et al., 2014). Aquests processos catabòlics provoquen l'alliberament d'aminoàcids al torrent sanguini (valina, leucina, etc) que podem veure reflectits en els pacients amb *cachexia* en el heatmap (majors concentracions, colors allunyats del blau). De la mateixa manera s'observa major presència d'intermedis com 3-hydroxybutyrate, producte de l'oxidació de lípids en contextos de dèficit energètic. A més, observem un augment de quilonate que podria reflectir a l'activació de la via del triptòfan associada a l'estrès inflamatori i oxidatiu, habitual en pacients amb cachexia (Faeron et al., 2011).

De la mateixa manera, tot i que no es veu tant al heatmap, podem observar en l'anàlisis univariant que la *Creatine* també presenta diferències en les concentracions segons el grup al que pertany el pacient. Aquests resultats també tenen coherència amb la fisiopatologia del síndrome, ja que un dels símptomes més rellevants de *Cachexia* comporta un elevat catabolisme proteic i muscular que es pot traduïr a un augment de les concentracions extracel · lulars de creatina i, en conseqüència, un augment en la concentració de creatina en la orina dels pacients.

CONCLUSIÓ

L'anàlisi estadístic exploratori que s'ha realitzat sobre les dades metabolòmiques del dataset *human_cachexia* ha permès identificar patrons associats a símptomes del desenvolupament del síndrome *Cachexia*. Els resultats han mostrat que una gran part dels metabòlits analitzats en la orina dels pacients presenten diferències significatives (anàlisi no paramètric) entre pacients amb *cachexia* i pacients control. Aquests resultats reforcen la hipòtesi que aquest perfil metabolòmic pot reflectir la identificació del mateix síndrome analitzant les alteracions en les concentracions d'aquests metabòlits. Els anàlisis multivariants (PCA, clustering i HEatmap) van poder ser-nos d'ajuda per a identificar aquests patrons, mostrant agrupaments de les mostres segons el grup i observant com cada grup tenia, en general, concentracions de metabòlits diferents. Per tant, es podria dir que els resultats obtinguts suggereixen que utilitzar aquest perfil metabolòmic per l'identificació de pacients amb *cachexia* podria ser una eina molt útil en el futur. Tot i així caldria aprofundir l'estudi i analitzar la seva aplicabilitat clínica en medicina.

REFERÈNCIES

- Evans, W.J., Morley, J.E., Argilés, J., Bales, C., Baracos, V., Guttridge, D., Jatoi, A., Kalantar-Zadeh, K., Lochs, H., Mantovani, G., Marks, D., Mitch, W.E., Muscaritoli, M., Najand, A., Ponikowski, P., Rossi Fanelli, F., Schambelan, M., Schols, A., Schuster, M., Thomas, D., Wolfe, R., & Anker, S.D. (2008). Cachexia: A new definition. *Clinical Nutrition*, 27(6), 793–799. https://doi.org/10.1016/j.clnu.2008.06.013

- Fearon, K., Strasser, F., Anker, S.D., Bosaeus, I., Bruera, E., Fainsinger, R.L., Jatoi, A., Loprinzi, C., MacDonald, N., Mantovani, G., Davis, M., Muscaritoli, M., Ottery, F., Radbruch, L., Ravasco, P., Walsh, D., Wilcock, A., Kaasa, S., & Baracos, V.E. (2011). Definition and classification of cancer cachexia: An international consensus. *The Lancet Oncology*, 12(5), 489–495. https://doi.org/10.1016/S1470-2045(10)70218-7

- Bioconductor (2024). *SummarizedExperiment: SummarizedExperiment Container*. Disponible a: https://bioconductor.org/packages/release/bioc/html/SummarizedExperiment.html

- Kassambara, A. (2020). *Practical Guide to Cluster Analysis in R: Unsupervised Machine Learning*. Datanovia. Disponible a: https://www.datanovia.com/en/lessons/cluster-analysis-in-r/

- ASP Teaching (2024). *Anàlisi Multivariant de Casos en R*. Disponible a: https://aspteaching.github.io/AMVCasos/

- MixOmics Team (2024). *Multivariate Analysis and Integration of Omics Data*. Disponible a: https://mixomicsteam.github.io/mixOmics-Vignette/

- Universitat Oberta de Catalunya (2024). *Exploració multivariant de dades òmiques*. Materials docents del Màster Universitari d'Anàlisi de Dades, UOC.

- Penet, M.F., Krishnamachary, B., Mironchik, Y., Wildes, F., Poussaint, T.Y., Lisok, A., et al. (2018). Predicting cancer-associated muscle wasting from urinary metabolomics. *Metabolomics*, 14(8), 1–13. https://doi.org/10.1007/s11306-018-1405-2

- Kassambara, A. (2017). *HCPC - Hierarchical Clustering on Principal Components Essentials*. STHDA. Disponible a: https://www.sthda.com/english/articles/31-principal-component-methods-in-r-practical-guide/117-hcpc-hierarchical-clustering-on-principal-components-essentials/