PAC 1 - Anàlisi de Dades Òmiques

Josep Rocaspana Codana

2024-10-25

Contents

Resum Executiu

En aquesta PEC, es realitza una anàlisi exploratòria d'un dataset metabolòmic. En concret, s'ha fet servir el dataset human_cachexia el qual conté dades de 77 pacients oncològics als quals se'ls classifica en dos grups: pacients control o caquèxics. Per a executar la tasca, s'utilitza el format SummarizedExperiment per a poder organitzar i estructurar les mostres, metabòlits i metadades associades amb la finalitat de poder tenir un maneig optimitzat de les dades.

El procés es va iniciar amb la generació del contenidor SummarizedExperiment, el qual inclou les dades de les concentracions dels metabòlits, informació sobre les mostres (Com la identificació del pacient o la seva condició muscular) i, finalment, una descripció de la metadata d'aquest SE.

Objectius

L'objectiu principal d'aquesta PAC és, doncs, explorar les diferències metabòliques entre pacients oncològics control i pacients oncològics amb la condició de caquèxia aprenent a generar contenidors com Summarized-Experiment i treballant amb una anàlisi estadística descriptiva i exploratòria.

Material i mètodes

Les dades amb les quals s'ha treballat corresponen al dataset human_cachexia, el qual pot trobar-se a la plataforma MetaboAnalyst. Aquest dataset conté diferents determinacions de metabòlits en mostres d'orina de 77 pacients oncològics els quals estan dividits entre pacients controls i pacients caquèxics. Aquest fet permet la possibilitat d'estudiar les diferències entre aquests grups.

En relació a les eines utilitzades, per a l'anàlisi i processament de les dades s'han fet servir diverses eines i paquets de R. Alguns dels recursos utilitzat al llarg de l'elaboració de la PAC són:

1) SummarizedExperiment: El qual ens permet crear i manipular el contenidor de dades, la qual cosa facilita la manipulació de les dades i metadades.

2) ggplot2: És un recurs bioinformàtic que permet la creació de gràfics i d'altres visualitzacions entre les quals inclou: boxplots, histogrames, gràfics de components principals (PCA), etc.

3) dplyr i tidyr: Permeten una manipulació i transformació de les dades de forma senzilla.

4) BiocManager i S4Vectors: Aquestes eines permeten la facilitació de la integració i compatibilitat de les estructures de dades per a R.

En quant al procediment de l'anàlisi, consta dels següents passos:

1) Generació del SummarizedExperiment (Organització de les dades): Les dades han sigut estructurades i contingudes a un contenidor SummarizedExperiment (se). Aquest contenidor presenta la matriu numèrica a la qual hi són les determinacions de les concentracions de metabòlits, les metadades relacionades amb les covariants de cada pacient (és a dir, Muscle.loss o estat muscular i Patient.ID [identificador]) i, també, els noms dels metabòlits.

2) Anàlisi exploratòria. Aquesta anàlisi ha constat de les següents parts:

- Determinació les dimensions de l'objecte se (número de metabòlits i mostres)
- Determinació dels noms dels elements de l'objecte se.
- Resum estadístic de les determinacions metabòliques mitjançant el càlcul de diferents estadístics (mínim, Q1, mediana, mitjana aritmètica, Q3, màxim i desviació estàndar)
- Visualització de les distribucions dels metabòlits en funció de l'estat muscular del pacient mitjançant boxplots.
- Anàlisi de Components Principals amb la finalitat de poder avaluar el comportament de cada grup i determinar la separació entre aquest en funció dels seus perfils metabòlics.
- Generació de matriu de correlació. Es calcula la matriu de correlació entre metabòlits i es visualitza mitjançant un heatmap amb la finalitat d'explorar relacions o associacions funcionals entre ells.
- Clustering jeràrquic. Es varen generar dos dendogrames de clústering jeràrquic, un per mostra i un per metabòlit amb la finalitat de poder detectar agrupacions a les nostres dades.

Resultats

Primerament, carrego les llibreries i paquets necessaris per a la realització de la tasca.A més a més, carrego les nostres dades i les visualitzo inicialment per a veure que ho he fet correctament.

```
library(dplyr)
```

```
##
## Adjuntando el paquete: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(ggplot2)

dataset_caquexia <- read.csv("C:/Users/Josep/Downloads/human_cachexia.csv")
head(dataset_caquexia)
```

```
##    Patient.ID Muscle.loss X1.6.Anhydro.beta.D.glucose X1.Methylnicotinamide
## 1     PIF_178    cachexic                       40.85                 65.37
## 2     PIF_087    cachexic                       62.18                340.36
## 3     PIF_090    cachexic                      270.43                 64.72
## 4 NETL_005_V1    cachexic                      154.47                 52.98
## 5     PIF_115    cachexic                       22.20                 73.70
## 6     PIF_110    cachexic                      212.72                 31.82
##   X2.Aminobutyrate X2.Hydroxyisobutyrate X2.Oxoglutarate X3.Aminoisobutyrate
## 1            18.73                 26.05           71.52              1480.30
## 2            24.29                 41.68           67.36               116.75
## 3            12.18                 65.37           23.81                14.30
## 4           172.43                 74.44         1199.91               555.57
## 5            15.64                 83.93           33.12                29.67
## 6            18.36                 80.64           47.94                17.46
##   X3.Hydroxybutyrate X3.Hydroxyisovalerate X3.Indoxylsulfate
## 1              56.83                 10.07            566.80
## 2              43.82                 79.84            368.71
## 3               5.64                 23.34            665.14
## 4             175.91                 25.03            411.58
## 5              76.71                 69.41            165.67
## 6              31.82                 35.16            183.09
##   X4.Hydroxyphenylacetate Acetate Acetone Adipate Alanine Asparagine Betaine
## 1                  120.30  126.47    9.49   38.09  314.19     159.17  109.95
## 2                  432.68  212.72   11.82  327.01  871.31     157.59  244.69
## 3                  292.95  314.19    4.44  131.63  464.05      89.12  116.75
## 4                  214.86   37.34  206.44  144.03  589.93     273.14  278.66
## 5                   97.51  407.48   44.26   15.03 1118.79      42.52  391.51
## 6                  132.95   81.45   14.44   25.28  237.46     157.59   66.69
##   Carnitine  Citrate Creatine Creatinine Dimethylamine Ethanolamine Formate
## 1    265.07  3714.50   196.37   16481.60        632.70       645.48  441.42
## 2    120.30  2617.57   212.72   15835.35        607.89       487.85  252.14
## 3     25.03   862.64   221.41   24587.66        735.10       407.48  249.64
## 4    200.34 13629.61    85.63   20952.22       1064.22       820.57  468.72
## 5     84.77   854.06   105.64    6768.26        242.26       365.04  114.43
## 6     40.04  1958.63   200.34   15677.78        614.00       459.44  314.19
##   Fucose Fumarate Glucose Glutamine Glycine Glycolate Guanidoacetate Hippurate
## 1 336.97     7.69  395.44    871.31 2038.56    685.40         154.47   4582.50
## 2 198.34    18.92 8690.62    601.85 1107.65    651.97         109.95   1737.15
## 3 186.79     7.10 1352.89    301.87  620.17    141.17         183.09   4315.64
## 4 407.48    96.54  862.64   1685.81 5064.45     70.81         102.51    757.48
## 5  26.05    19.69 6836.29    432.68  395.44     26.58          52.98   1152.86
## 6 123.97     5.05  512.86    298.87  482.99    428.38          57.97   3568.85
##   Histidine Hypoxanthine Isoleucine Lactate Leucine Lysine Methylamine
## 1    925.19        97.51       5.58  106.70   42.10 146.94       52.46
## 2    845.56        82.27       8.17  368.71   77.48 284.29       23.57
## 3    284.29       114.43       9.30  749.95   31.50  97.51       18.73
## 4   1043.15       223.63      37.71  368.71  103.54 290.03       48.91
## 5    327.01        66.69      40.04 3640.95  101.49 122.73       27.94
```

```
## 6    459.44        62.80       8.17  113.30   28.79 120.30       36.97
##   Methylguanidine N.N.Dimethylglycine O.Acetylcarnitine Pantothenate
## 1            9.97               23.34             52.98        25.79
## 2            7.69               87.36             50.40       186.79
## 3            4.66               24.53              5.58       145.47
## 4          141.17               40.04            254.68        42.52
## 5            5.31               46.06             45.60        74.44
## 6           43.38               24.29             13.46        35.52
##   Pyroglutamate Pyruvate Quinolinate  Serine Succinate Sucrose Tartrate Taurine
## 1        437.03    21.12      165.67  284.29    154.47   45.15    97.51 1919.85
## 2        437.03    36.97       72.97  391.51    244.69  459.44    32.79 1261.43
## 3        713.37    29.37      192.48  295.89    142.59  160.77    16.28 4272.69
## 4        566.80    64.07       86.49 1248.88    144.03  111.05   837.15 1525.38
## 5        184.93    12.30       38.09  206.44     68.72   75.19     4.53  468.72
## 6        432.68    32.79      112.17  387.61     33.45  336.97    24.05 2059.05
##   Threonine Trigonelline Trimethylamine.N.oxide Tryptophan Tyrosine Uracil
## 1    184.93       943.88                2121.76     259.82   290.03 111.05
## 2    198.34       208.51                 639.06      83.10   167.34  46.99
## 3    109.95       192.48                1152.86      82.27    60.34  31.50
## 4    376.15       992.27                1450.99     235.10   323.76  30.57
## 5     64.07        86.49                 172.43     103.54   142.59  44.26
## 6    105.64       862.64                 880.07     239.85   127.74  29.67
##   Valine  Xylose cis.Aconitate myo.Inositol trans.Aconitate pi.Methylhistidine
## 1  86.49   72.24        237.46       135.64           51.94              157.59
## 2 109.95  192.48        333.62       376.15          217.02              307.97
## 3  59.15 2164.62        330.30        86.49           58.56              145.47
## 4 102.51  125.21       1863.11       247.15           75.94              249.64
## 5 160.77  186.79        101.49       749.95           98.49               84.77
## 6  36.97   89.12        287.15       129.02          121.51              399.41
##   tau.Methylhistidine
## 1              160.77
## 2              130.32
## 3               83.93
## 4              254.68
## 5               79.84
## 6               68.72
```

Una vegada carregat, veig que s'ha carregat correctament i procedeixo a la operació amb el fitxer

Després d'haver visualitzat el dataset, veig que consisteix en:

77 pacients amb els seus ID (primera columna) El seu estat muscular (caquèxic vs normal) *Les diferents determinacions de metabòlits (3a columna en endavant)

Així doncs, carrego els paquets i llibreries necessaris i començo a operar. Per a poder obtenir l'objecte SummarizedExperiment, hem de generar prèviament les seves parts. Per a facilitar la comprensió del que s'ha dut a terme i que tot el codi sigui present a un chunk, aniré segmentant els pasos 1 a 1 i marcaré al codi quin pas és quin.

- Pas 1: Primer, faig l'extracció de la matriu numèrica del meu dataset amb la finalitat de generar el assay. Amb aquest codi selecciono totes les columnes del dataset original des de la 3a fins la última columna (mitjançant ncol). Això ho faig perquè a assay_data només vull que contingui la matriu numèrica amb les determinacions de metabòlits. Posteriorment transposo la matriu amb la finalitat que cada columna correspongui a una mostra (un pacient) i cada fila representi un metabòlit (la qual cosa s'ajusta bé al format de SummarizedExperiment). Finalment, faig servir as.matrix per convertir les dades seleccionades en una matriu numèrica (ja que SummarizedExperiment espera aquest format.)

- Pas 2: Posteriorment, faig una assignació dels noms de les columnes i de les files. Primerament, fent servir colnames, el que faig es assignar els identificadors dels pacients com els noms de les columnes ja que cada columna representa una mostra única.
- Pas 3:També assigno els noms dels metabòlits com a noms de les files per a assay_data. Aquests es troben de la columna 3 cap endavant
- Pas 4:Quan ja hem generat el assay_data definim pacients_metadata, creo un data.frame amb aquest nom amb la finalitat d'emmagatzemar les metadades de les mostres, que en el cas d'aquest dataset, inclouen les covariants de ID (Patient.ID) i la seva condició muscular (Muscle.loss). Així doncs, aquest dataframe será el colData del meu SummarizedExperiment i guardarà la informació de cada mostra

Selecciono les columnes Patient.ID i Muscle.loss perquè el ID serveix com a indicador únic de cada pacient i la variable Muscle.loss es una variable que indica quina condició té cada pacient, i serà vital per al anàlisi (encara que sigui merament exploratori), ja que s'utilitzarà per a comparar els grups.

Posteriorment, faig una assignació del nom de files, assignant els ID com noms de fila a sample_metadata amb la finalitat que cada fila de ColData es correspongui adequadament amb les columnes del assay_data. Així doncs, aconseguim l'alineament de les metadates amb els metabòlits. En última instància, converteixo en dataframe perquè sigui compatible amb SummarizedExperiment.

- Pas 5: Posteriorment, escullo els noms de les columnes de la tercera fins la ultima del dataset original que correspon als metabolits i creo una metadata que només tingui els noms dels metabòlits. Així doncs em queda una llista dels noms dels metabòlits. No assigno rownames per a evitar les duplicacions, ja que els noms dels metabòlits ja es troben a la columna Metabolite. Aquest dataframe doncs, serveix com a rowData en sí mateix en el SummarizedExperiment. Finalment, converteixo el pacients_metadata a un dataframe (ja que SummaryExperiment ho requereix)
- Pas 6: Així doncs, genero l'objecte se (SummarizedExperiment) que conté:

1) la matriu numèrica de dades on s'emmagatzemen els valors de les determinacions de metabòlits (assays)
2) Les metadades de les mostres (colData), que descriu les característiques de cada pacient.
3) Metadades de les característiques (rowData) que descriu cada metabòlit del dataset
4) Informació addicional que he pogut extreure anant a "https://www.metaboanalyst.ca/MetaboAnalyst/upload/PowerUploadView.xhtml" per a la descripció del dataset

i, finalment, imprimeixo l'objecte se per a la seva visualització.

```
if (!requireNamespace("BiocManager", quietly = TRUE)) {
  install.packages("BiocManager")
}

if (!requireNamespace("SummarizedExperiment", quietly = TRUE)) {
  BiocManager::install("SummarizedExperiment")
}

if (!requireNamespace("S4Vectors", quietly = TRUE)) {
  BiocManager::install("S4Vectors")
}

library(SummarizedExperiment)
```

```
## Cargando paquete requerido: MatrixGenerics
```

```
## Cargando paquete requerido: matrixStats

##
## Adjuntando el paquete: 'matrixStats'

## The following object is masked from 'package:dplyr':
##
##     count

##
## Adjuntando el paquete: '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

## Cargando paquete requerido: GenomicRanges

## Cargando paquete requerido: stats4

## Cargando paquete requerido: BiocGenerics

##
## Adjuntando el paquete: 'BiocGenerics'

## The following objects are masked from 'package:dplyr':
##
##     combine, intersect, setdiff, union

## 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, table,
##     tapply, union, unique, unsplit, which.max, which.min
```

```
## Cargando paquete requerido: S4Vectors

##
## Adjuntando el paquete: 'S4Vectors'

## The following objects are masked from 'package:dplyr':
##
##     first, rename

## The following object is masked from 'package:utils':
##
##     findMatches

## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname

## Cargando paquete requerido: IRanges

##
## Adjuntando el paquete: 'IRanges'

## The following objects are masked from 'package:dplyr':
##
##     collapse, desc, slice

## The following object is masked from 'package:grDevices':
##
##     windows

## Cargando paquete requerido: GenomeInfoDb

## Cargando paquete requerido: Biobase

## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.

##
## Adjuntando el paquete: 'Biobase'

## The following object is masked from 'package:MatrixGenerics':
##
##     rowMedians

## The following objects are masked from 'package:matrixStats':
##
##     anyMissing, rowMedians
```

```r
library(S4Vectors)

assay_data <- as.matrix(t(dataset_caquexia[, 3:ncol(dataset_caquexia)])) #pas 1

colnames(assay_data) <- dataset_caquexia$Patient.ID #pas 2

rownames(assay_data) <- colnames(dataset_caquexia)[3:ncol(dataset_caquexia)] #pas3


pacients_metadata <- data.frame(
  Patient.ID = dataset_caquexia$Patient.ID,
  Muscle.loss = dataset_caquexia$Muscle.loss
)

rownames(pacients_metadata) <- pacients_metadata$Patient.ID
pacients_metadata <- DataFrame(pacients_metadata) #PAS 4



metabolite_metadata <- DataFrame(Metabolite = colnames(dataset_caquexia)[3:ncol(dataset_caquexia)]) #pa



se <- SummarizedExperiment(
  assays = list(counts = assay_data),
  colData = pacients_metadata,
  rowData = metabolite_metadata,
  metadata = list(
    descripcio_dataset = "Dades metabològiques de 77 pacients caquèxics i controls en un estudi pilot m
    informacio_columna = "Cada columna representa una mostra (és a dir, un pacient) i la seva condició
    informacio_fila = "Cada fila representa un metabòlit determinat a les mostres"
    )
)

print(se)      #pas6
```

```
## class: SummarizedExperiment
## dim: 63 77
## metadata(3): descripcio_dataset informacio_columna informacio_fila
## assays(1): counts
## rownames(63): X1.6.Anhydro.beta.D.glucose X1.Methylnicotinamide ...
##   pi.Methylhistidine tau.Methylhistidine
## rowData names(1): Metabolite
## colnames(77): PIF_178 PIF_087 ... NETL_003_V1 NETL_003_V2
## colData names(2): Patient.ID Muscle.loss
```

Anàlisi exploratòria

Després d'haver generat el SummarizedExperiment, podem procedir amb l'anàlisi exploratòria de les dades carregant les llibreries necessàries

Primerament, determinem:

1) Les dimensions de l'objecte se
2) Els noms dels elements de l'objecte se.

Aquests dos paràmetres ens donen informació sobre paràmetres bàsics com el número de files, columnes i els noms dels elements del nostre objecte.

```
library(SummarizedExperiment)
library(ggplot2)
library(dplyr)

metabolite_data <- assay(se)
sample_metadata <- colData(se)

#Determinem les dimensions de l'objecte se (número de metabòlits i mostres)
dim(se)
```

```
## [1] 63 77
```

```
#determinem els noms dels elements del SummarizedExperiment
names(se)
```

```
##  [1] "X1.6.Anhydro.beta.D.glucose" "X1.Methylnicotinamide"
##  [3] "X2.Aminobutyrate"            "X2.Hydroxyisobutyrate"
##  [5] "X2.Oxoglutarate"             "X3.Aminoisobutyrate"
##  [7] "X3.Hydroxybutyrate"          "X3.Hydroxyisovalerate"
##  [9] "X3.Indoxylsulfate"           "X4.Hydroxyphenylacetate"
## [11] "Acetate"                     "Acetone"
## [13] "Adipate"                     "Alanine"
## [15] "Asparagine"                  "Betaine"
## [17] "Carnitine"                   "Citrate"
## [19] "Creatine"                    "Creatinine"
## [21] "Dimethylamine"               "Ethanolamine"
## [23] "Formate"                     "Fucose"
## [25] "Fumarate"                    "Glucose"
## [27] "Glutamine"                   "Glycine"
## [29] "Glycolate"                   "Guanidoacetate"
## [31] "Hippurate"                   "Histidine"
## [33] "Hypoxanthine"                "Isoleucine"
## [35] "Lactate"                     "Leucine"
## [37] "Lysine"                      "Methylamine"
## [39] "Methylguanidine"             "N.N.Dimethylglycine"
## [41] "O.Acetylcarnitine"           "Pantothenate"
## [43] "Pyroglutamate"               "Pyruvate"
## [45] "Quinolinate"                 "Serine"
## [47] "Succinate"                   "Sucrose"
## [49] "Tartrate"                    "Taurine"
## [51] "Threonine"                   "Trigonelline"
## [53] "Trimethylamine.N.oxide"      "Tryptophan"
## [55] "Tyrosine"                    "Uracil"
## [57] "Valine"                      "Xylose"
## [59] "cis.Aconitate"               "myo.Inositol"
## [61] "trans.Aconitate"             "pi.Methylhistidine"
## [63] "tau.Methylhistidine"
```

```r
#Generem un resum de les dades d'expressió (assay) amb les estadístiques de cada metabòlits en general

metabolite_summary <- data.frame(
  Minim = apply(assay(se), 1, min),
  Q1 = apply(assay(se), 1, quantile, probs = 0.25),
  mediana = apply(assay(se), 1, median),
  mitjana = apply(assay(se), 1, mean),
  Q3 = apply(assay(se), 1, quantile, probs = 0.75),
  Maxim = apply(assay(se), 1, max),
  Desviacio_estandar = apply(assay(se), 1, sd)
)

cat("Resum estadístic dels valors generals dels metabòlits:")
```

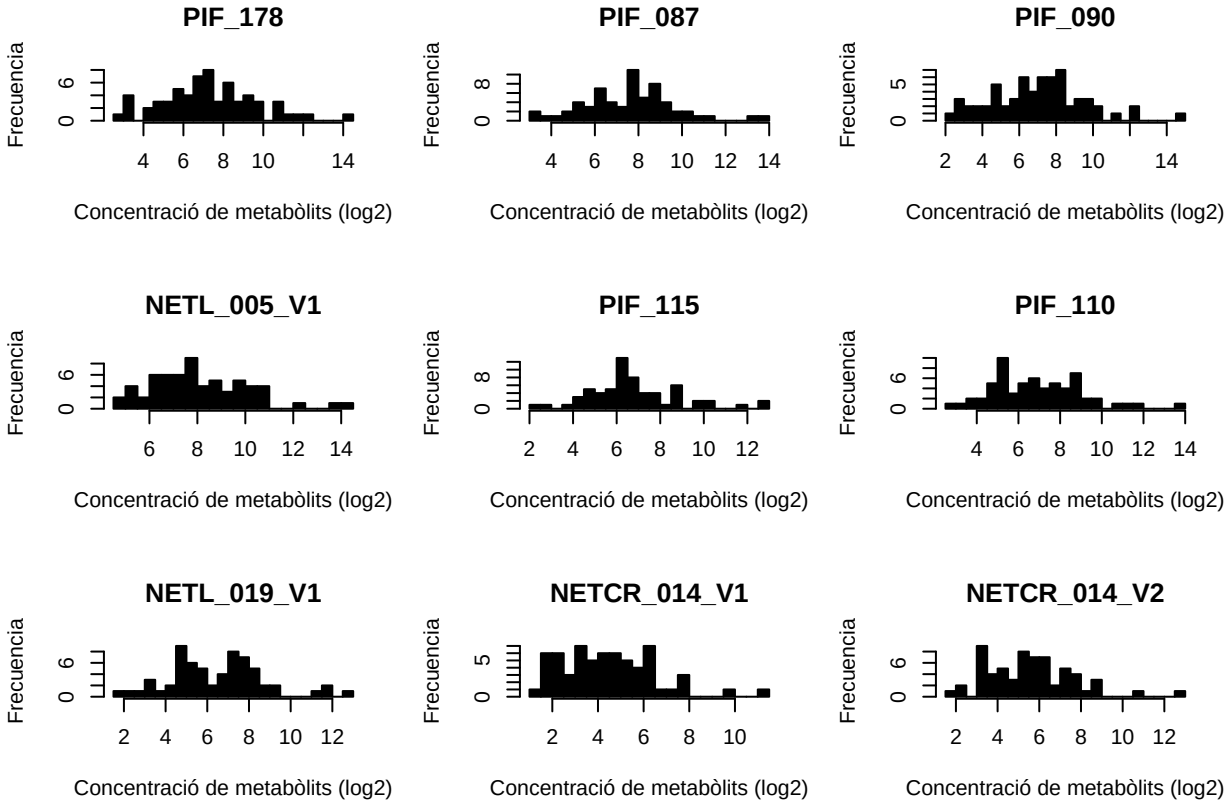
```
## Resum estadístic dels valors generals dels metabòlits:
```

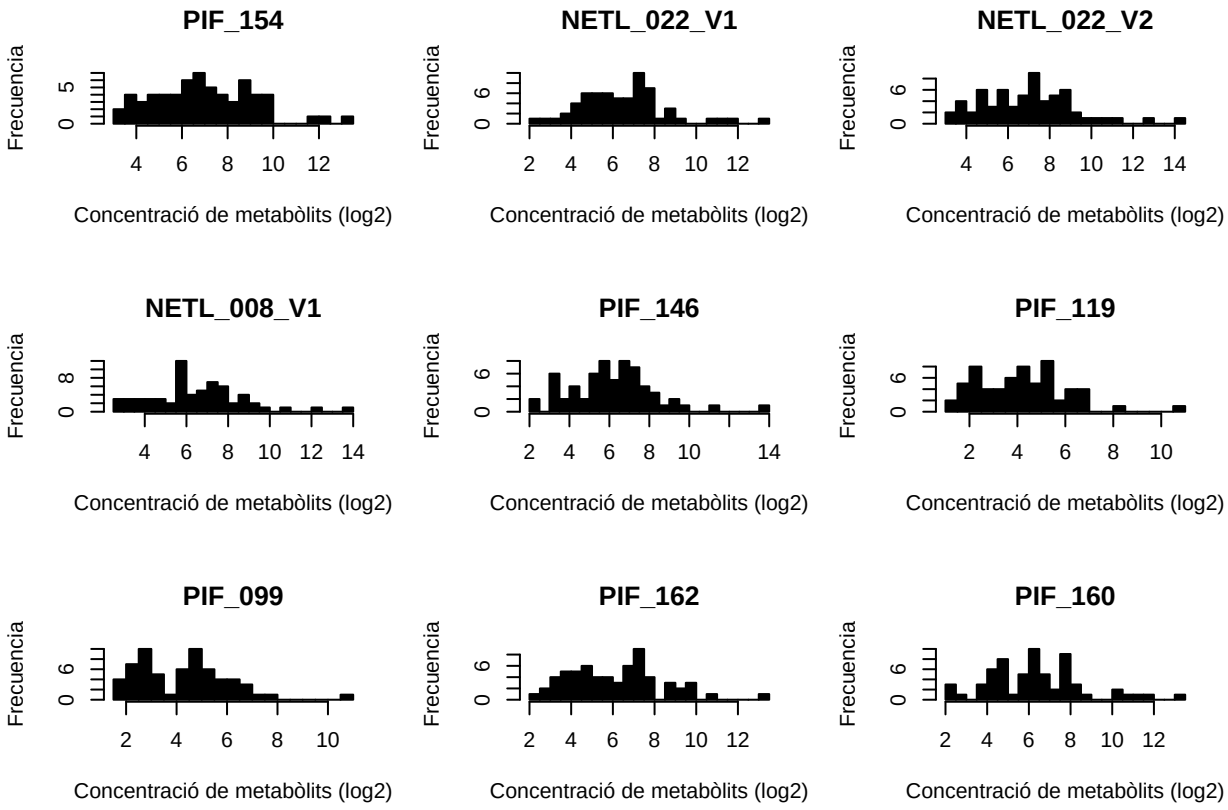
```r
print(head(metabolite_summary))
```

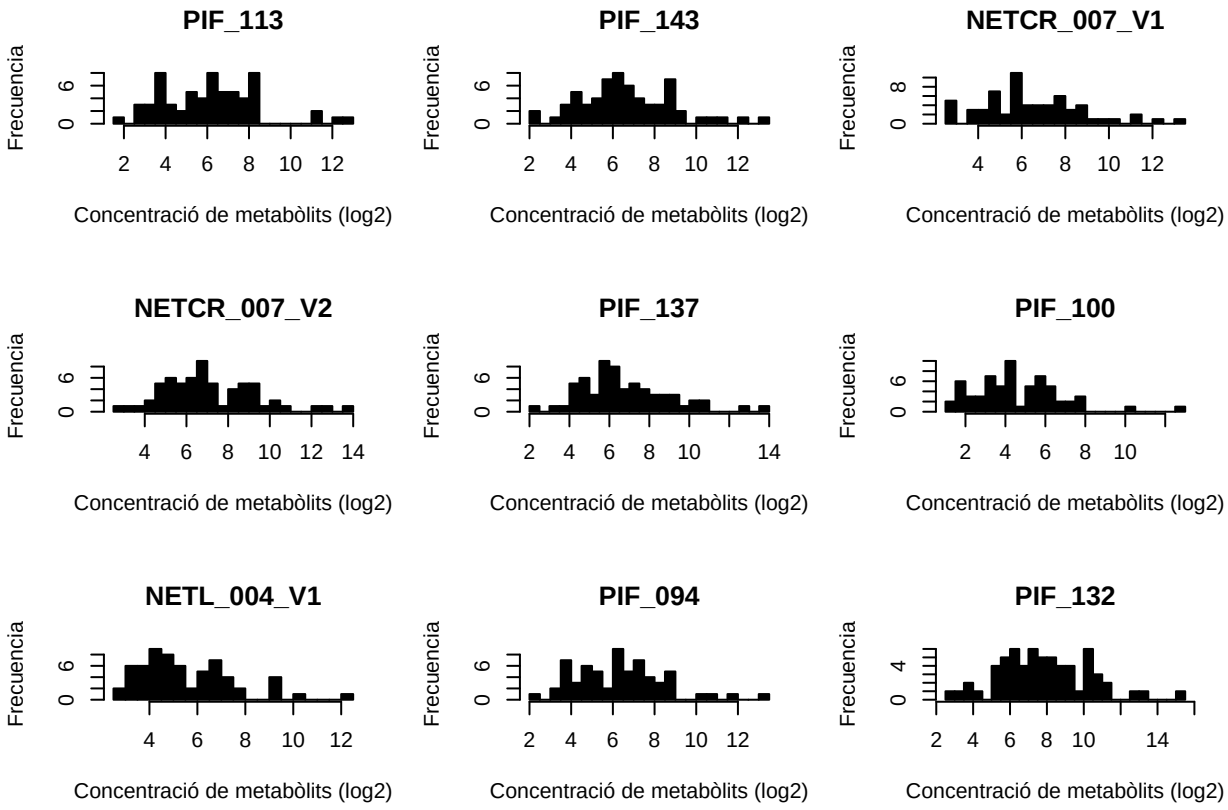
```
##                              Minim    Q1 mediana   mitjana     Q3   Maxim
## X1.6.Anhydro.beta.D.glucose   4.71 28.79   45.60 105.63039 141.17  685.40
## X1.Methylnicotinamide         6.42 15.80   36.60  71.57364  73.70 1032.77
## X2.Aminobutyrate              1.28  5.26   10.49  18.15974  19.49  172.43
## X2.Hydroxyisobutyrate         4.85 15.80   32.46  37.25065  54.60   93.69
## X2.Oxoglutarate               5.53 22.42   55.15 145.08714  92.76 2465.13
## X3.Aminoisobutyrate           2.61 11.70   22.65  76.75636  56.26 1480.30
##                              Desviacio_estandar
## X1.6.Anhydro.beta.D.glucose          130.02560
## X1.Methylnicotinamide                133.19281
## X2.Aminobutyrate                      27.61453
## X2.Hydroxyisobutyrate                 23.95681
## X2.Oxoglutarate                      342.52217
## X3.Aminoisobutyrate                  191.01424
```

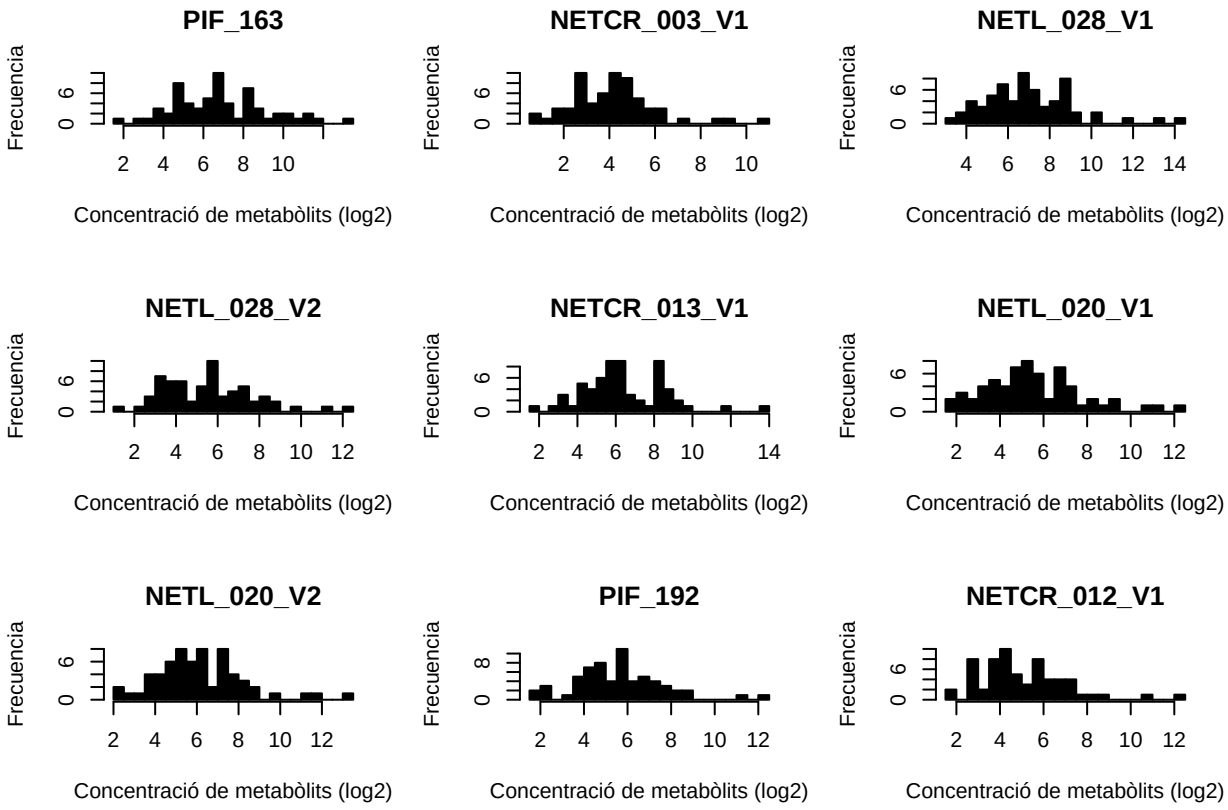
Ara podem fer una inspecció visual dels metabòlits a nivell de histograma. Degut a la natura de les dades metabolòmiques, és molt habitual que hi hagi valors molt baixos i d'altres que siguin molt, molt més elevats. Per aquesta raó, abans de graficar amb histogrames les dades relacionades amb els metabòlits, decideixo aplicar una transformació logaritmica per a millorar la seva visualització. A més a més, li sumo una unitat per a evitar el log(0)

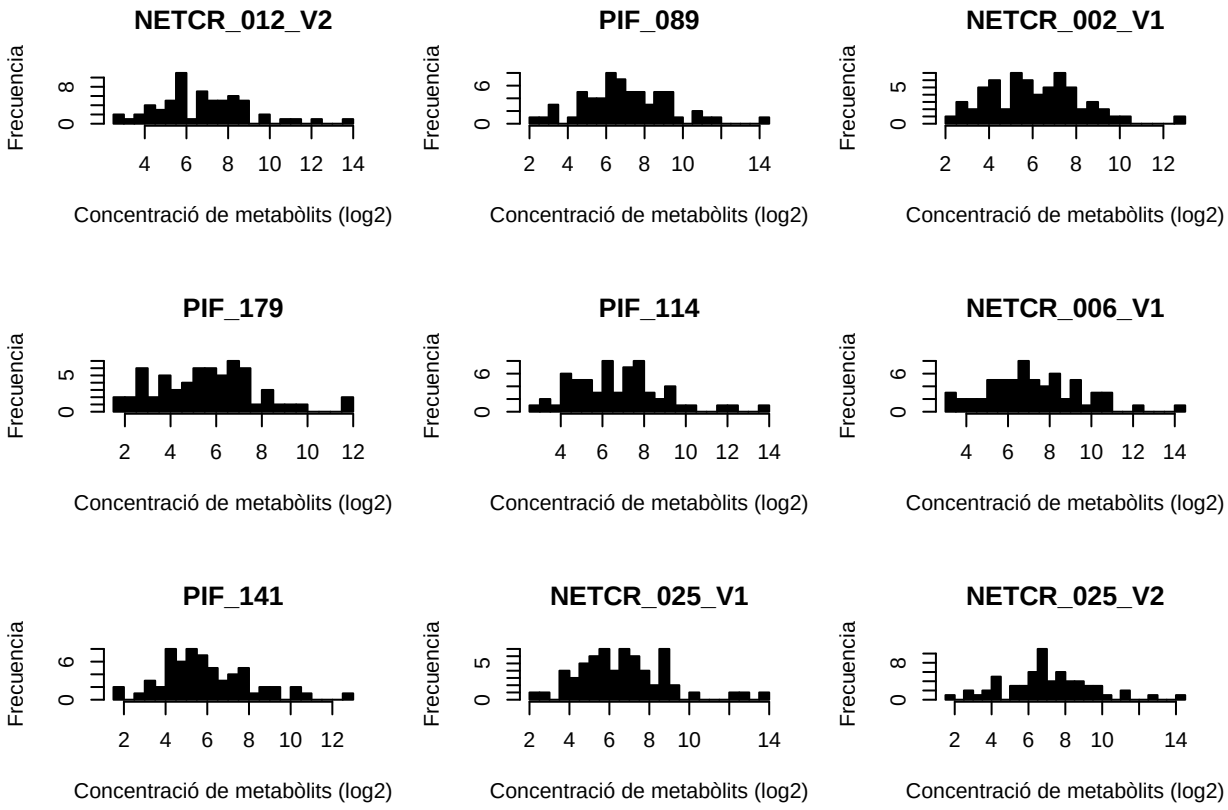
```r
dades_metabolits_logaritmiques <- log2(assay(se) + 1)
grafica_configuracio <- par(mfrow = c(3, 3))

#Grafiquem l'histograma de cada mostra fent servir les dades transformades logarítmicament.

for (i in 1:ncol(dades_metabolits_logaritmiques)) {
    hist(dades_metabolits_logaritmiques[, i],
         main = colnames(dades_metabolits_logaritmiques)[i],
         xlab = "Concentració de metabòlits (log2)",
         ylab = "Frecuencia",
         col = "black",
         breaks = 20)
}
```

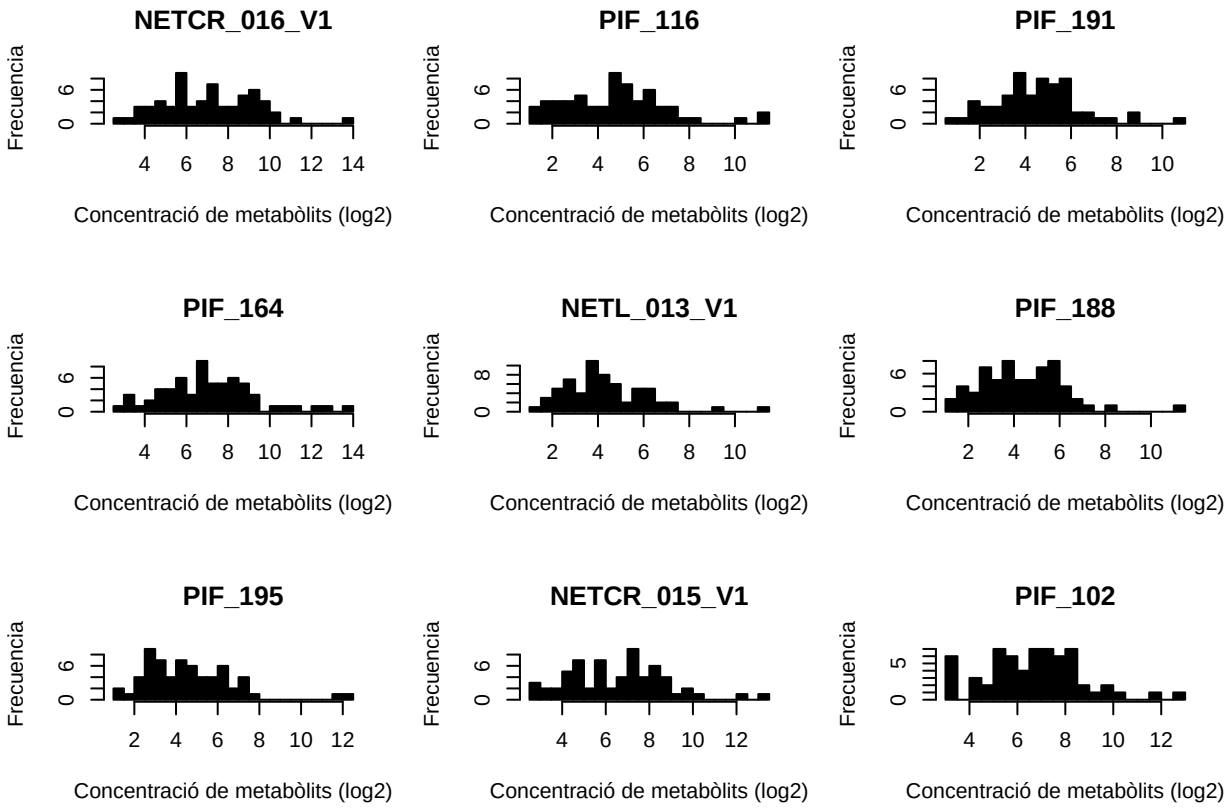

PIF_178
Frecuencia
0 6
4 6 8 10 14
Concentració de metabòlits (log2)
PIF_087
Frecuencia
0 8
4 6 8 10 12 14
Concentració de metabòlits (log2)
PIF_090
Frecuencia
0 5
2 4 6 8 10 14
Concentració de metabòlits (log2)
NETL_005_V1
Frecuencia
0 6
6 8 10 12 14
Concentració de metabòlits (log2)
PIF_115
Frecuencia
0 8
2 4 6 8 10 12
Concentració de metabòlits (log2)
PIF_110
Frecuencia
0 6
4 6 8 10 12 14
Concentració de metabòlits (log2)
NETL_019_V1
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)
NETCR_014_V1
Frecuencia
0 5
2 4 6 8 10
Concentració de metabòlits (log2)
NETCR_014_V2
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)

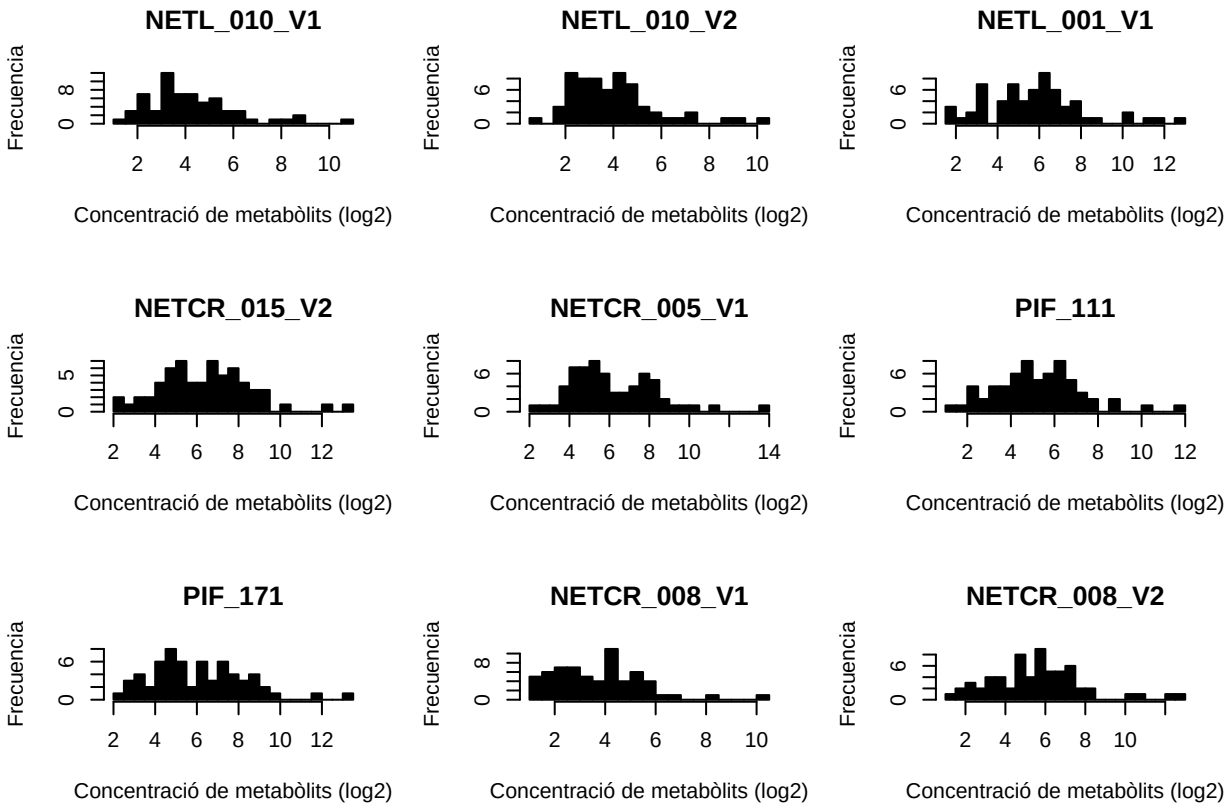

PIF_154
Frecuencia
0 5
4 6 8 10 12
Concentració de metabòlits (log2)
NETL_022_V1
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)
NETL_022_V2
Frecuencia
0 6
4 6 8 10 12 14
Concentració de metabòlits (log2)
NETL_008_V1
Frecuencia
0 8
4 6 8 10 12 14
Concentració de metabòlits (log2)
PIF_146
Frecuencia
0 6
2 4 6 8 10 14
Concentració de metabòlits (log2)
PIF_119
Frecuencia
0 6
2 4 6 8 10
Concentració de metabòlits (log2)
PIF_099
Frecuencia
0 6
2 4 6 8 10
Concentració de metabòlits (log2)
PIF_162
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)
PIF_160
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)

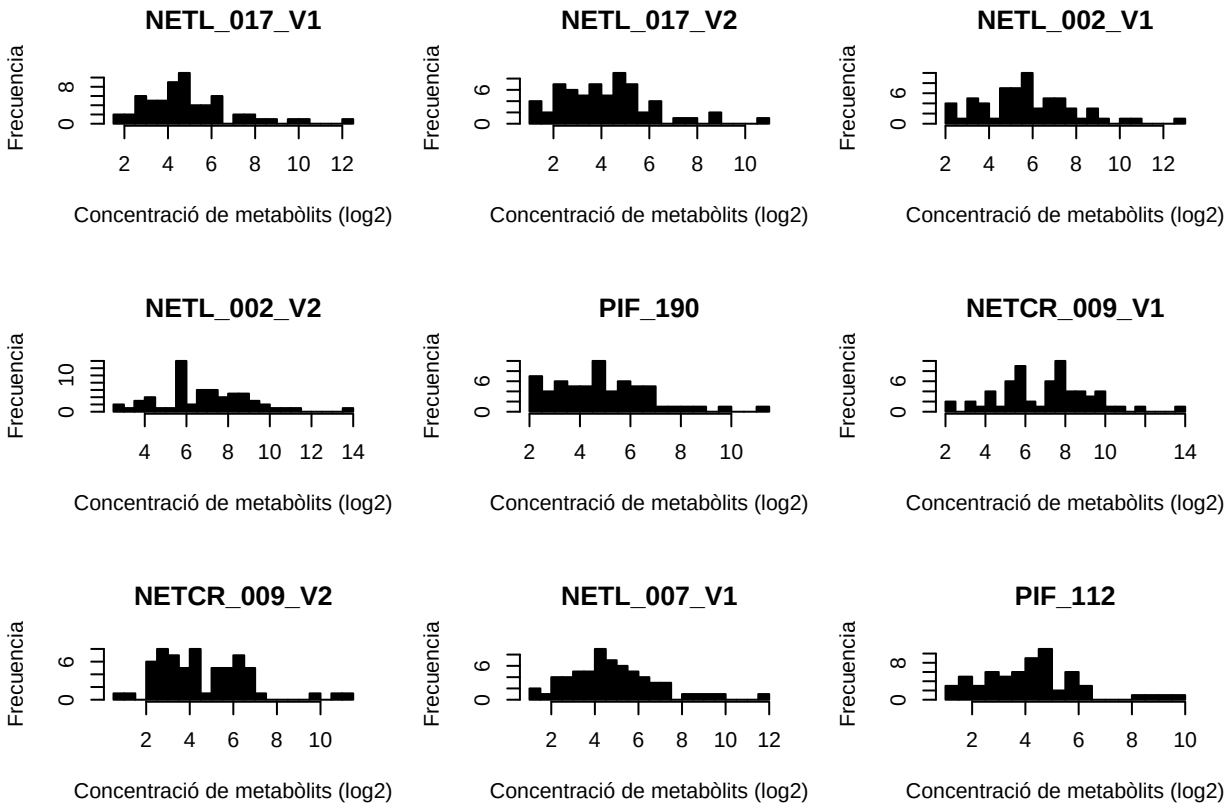

PIF_113
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)
PIF_143
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)
NETCR_007_V1
Frecuencia
0 8
4 6 8 10 12
Concentració de metabòlits (log2)
NETCR_007_V2
Frecuencia
0 6
4 6 8 10 12 14
Concentració de metabòlits (log2)
PIF_137
Frecuencia
0 6
2 4 6 8 10 14
Concentració de metabòlits (log2)
PIF_100
Frecuencia
0 6
2 4 6 8 10
Concentració de metabòlits (log2)
NETL_004_V1
Frecuencia
0 6
4 6 8 10 12
Concentració de metabòlits (log2)
PIF_094
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)
PIF_132
Frecuencia
0 4
2 4 6 8 10 14
Concentració de metabòlits (log2)


PIF_163
NETCR_003_V1
NETL_028_V1
NETL_028_V2
NETCR_013_V1
NETL_020_V1
NETL_020_V2
PIF_192
NETCR_012_V1
Frecuencia
Concentració de metabòlits (log2)


NETCR_012_V2
PIF_089
NETCR_002_V1
PIF_179
PIF_114
NETCR_006_V1
PIF_141
NETCR_025_V1
NETCR_025_V2
Frecuencia
Concentració de metabòlits (log2)


NETCR_016_V1
PIF_116
PIF_191
PIF_164
NETL_013_V1
PIF_188
PIF_195
NETCR_015_V1
PIF_102
Frecuencia
Concentració de metabòlits (log2)


NETL_010_V1
Frecuencia
0 8
2 4 6 8 10
Concentració de metabòlits (log2)
NETL_010_V2
Frecuencia
0 6
2 4 6 8 10
Concentració de metabòlits (log2)
NETL_001_V1
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)
NETCR_015_V2
Frecuencia
0 5
2 4 6 8 10 12
Concentració de metabòlits (log2)
NETCR_005_V1
Frecuencia
0 6
2 4 6 8 10 14
Concentració de metabòlits (log2)
PIF_111
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)
PIF_171
Frecuencia
0 6
2 4 6 8 10 12
Concentració de metabòlits (log2)
NETCR_008_V1
Frecuencia
0 8
2 4 6 8 10
Concentració de metabòlits (log2)
NETCR_008_V2
Frecuencia
0 6
2 4 6 8 10
Concentració de metabòlits (log2)

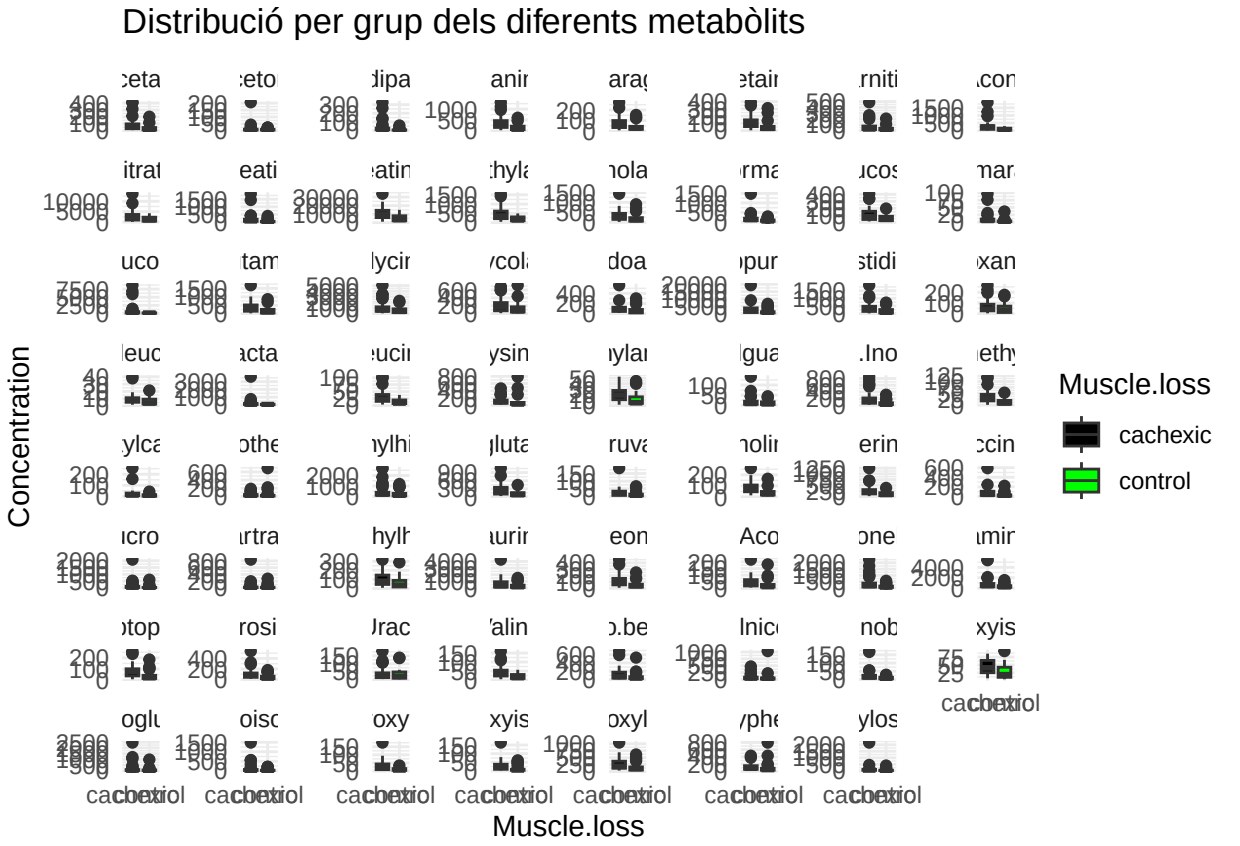



```
#Una altra possibilitat és veure com canvien les concentracions en funció del grup (control o caquèxic)
#Per a fer això, he decidit generar una comparació múltiple.



metabolite_df <- as.data.frame(t(metabolite_data))
metabolite_df$Muscle.loss <- sample_metadata$Muscle.loss


library(tidyr)
```

```
##
## Adjuntando el paquete: 'tidyr'

## The following object is masked from 'package:S4Vectors':
##
##     expand
```

```
metabolite_llarg <- metabolite_df %>%
  pivot_longer(cols = -Muscle.loss, names_to = "Metabolite", values_to = "Concentration")

ggplot(metabolite_llarg, aes(x = Muscle.loss, y = Concentration, fill = Muscle.loss)) +
  geom_boxplot() +
  facet_wrap(~ Metabolite, scales = "free_y") +
  scale_fill_manual(values = c("control" = "green", "cachexic" = "black")) +
```

```
  labs(title = "Distribució per grup dels diferents metabòlits") +
  theme_minimal()
```



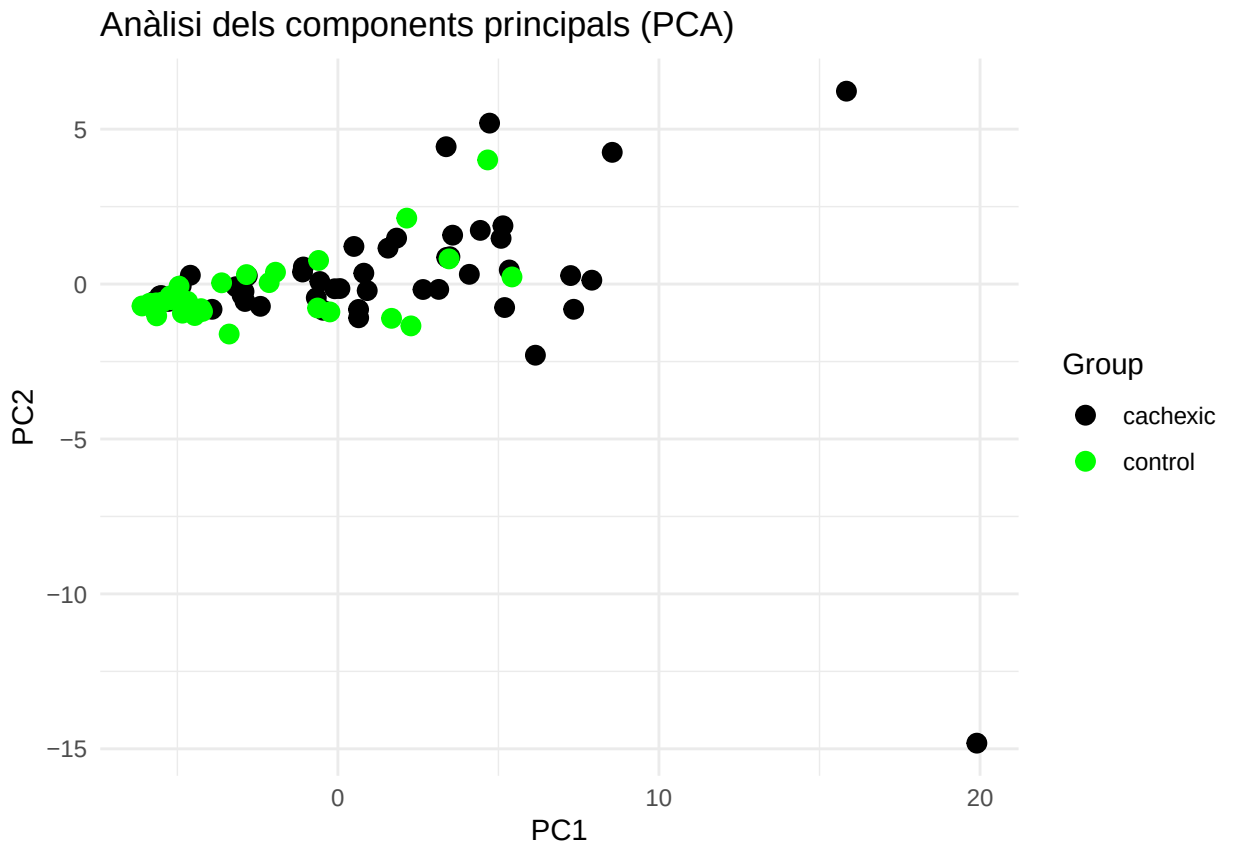
Per una banda, els histogrames amb les concentracions de metabòlits transformades logarítmicament permeten veure la distribució d'aquestes molècules per a cada pacient, permetent la visualització de patrons i variabilitat intraindividual entre aquests metabòlits. Aquesta representació NO diferencia entre grups i controls.

D'altra banda, al boxplot el que es pretén es representar les concentracions (NO logaritmiques) de diferents metabòlits en pacients controls i caquèxics. Gràcies a això, es pot veure de forma visual i ràpida les variacions de la concentració de metabòlits entre grups, donant una idea general d'aquesta variació. Com es pot observar, hi ha una una gran variabilitat entre metabòlits i una gran presència de valors extrems. Com podem veure, hi ha metabòlits com la creatinina, la leucina o la alanina que són marcadament més presents al grup de caquèxics en comparació al grup control. De fet, a grans trets, la concentració de tots aquests metabòlits en orina sembla ser superior en els pacients caquèxics.

Després d'haver visualitzat les dades, podem fer un anàlisi de components principals, com hem vist a la teoria, per a reduir la dimensionalitat de les dades i resumir la variabilitat en pocs components (PC1, PC2) els quals explican la major part de la variabilitat de les nostres dades.

```
analisi_pca <- prcomp(t(assay(se)), center = TRUE, scale. = TRUE)

dataframe_pca <- data.frame(analisi_pca$x, Group = colData(se)$Muscle.loss)

#Així doncs, grafiquem els dos components principals PC1, PC2.
ggplot(dataframe_pca, aes(x = PC1, y = PC2, color = Group)) +
  geom_point(size = 3) +
```

```
  labs(title = "Anàlisi dels components principals (PCA)", x = "PC1", y = "PC2") +
  scale_color_manual(values = c("control" = "green", "cachexic" = "black")) +
  theme_minimal()
```

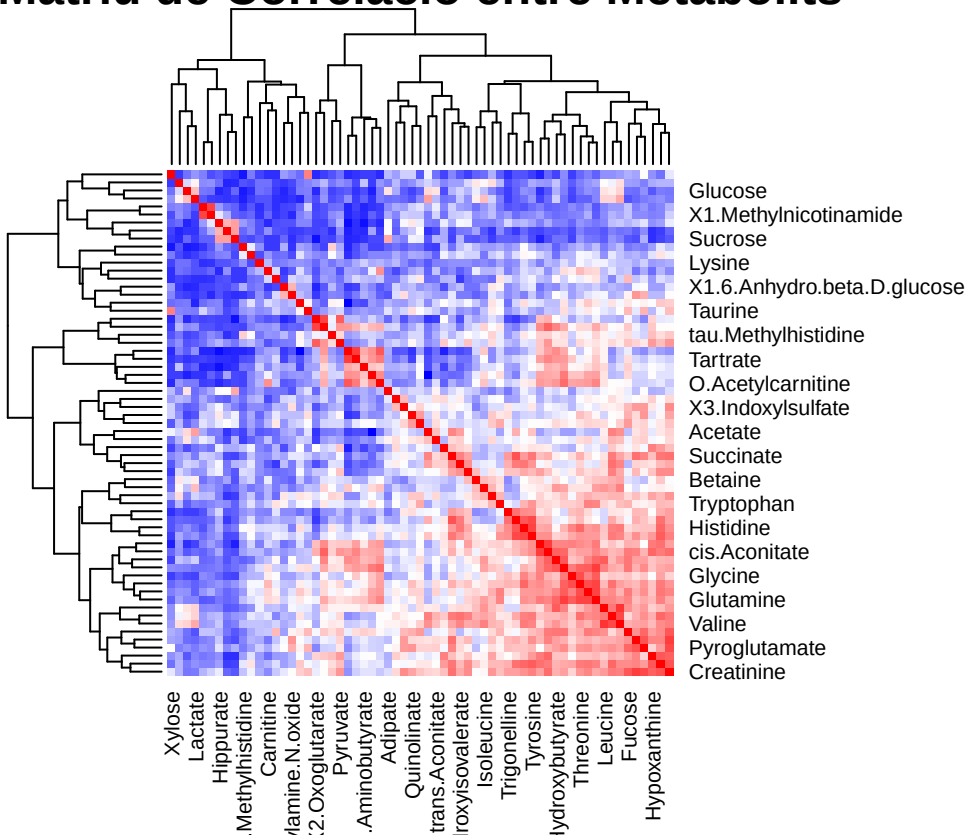


En el gràfic podem veure les mostres control representades en color verd, mentre que les mostres de pacients caquèxics estan en negre. Podem veure que hi ha cert solapament, no obstant això, alguns punts de pacients caquèxics mostren dispersió als extrems, la qual cosa podria indicar una variabilitat en el perfil metabòlic d'aquests pacients (això s'hauria de confirmar després estadísticament amb proves estadístiques). D'aquesta forma, aquest tipus d'anàlisi permet comprendre de forma visual diferencies entre grups.

Per a complementar aquest estudi exploratori de les dades, podem calcular i visualitzar la matriu de correlació entre els metabòlits. Aquesta matriu ens donarà informació sobre quins metabòlits tenen tendència a variar junts, fet que revela relacions entre metabòlits que ens poden interessar a l'hora d'analitzar perfils metabòlics.

Per exemple: una correlació elevada (positiva o negativa) entre alguns metabòlits podria suggerir que participen en vies metabòliques que estiguin relacionades. De fet, si s'aprofundís en l'anàlisi i es descobrissin metabòlits que mostren una alta correlació entre sí, podrien ser proposats com a candidats com a potencials biomarcadors per formar perfils metabòlics que puguin diferenciar als pacients amb càncer sense pèrdua muscular (control) d'aquells que tenen pèrdua muscular (caquèxia).

```
matriu_correlacio <- cor(t(assay(se)), use = "pairwise.complete.obs")
heatmap(matriu_correlacio, main ="Matriu de Correlació entre Metabòlits",
        col = colorRampPalette(c("blue", "white", "red")) (100),
        symm = TRUE)
```

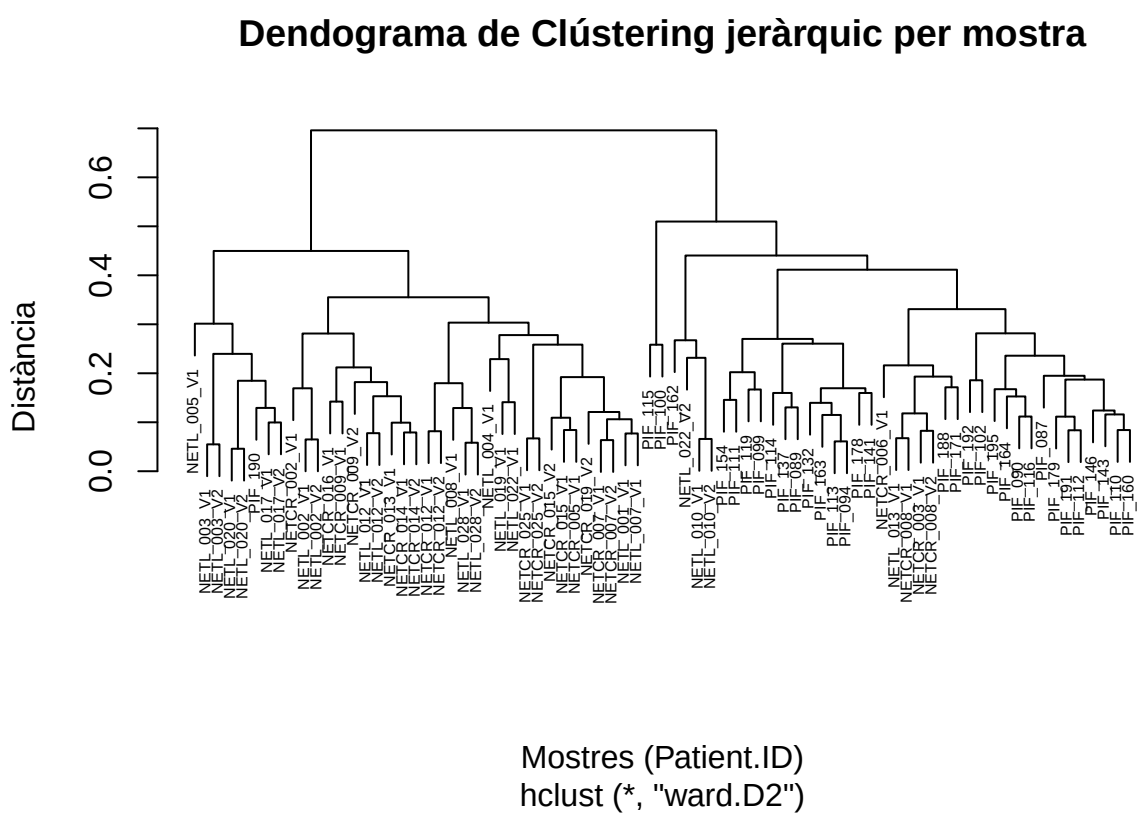


Només per mencionar alguna cosa relacionada amb els resultats obtinguts per la matriu, trobem alguns metabòlits a l'extrem inferior dret (valina, creatinina), que mostren correlacions positives amb metabòlits propers, la qual cosa pot indicar que estan relacionats funcionalment o que puguin participar.

Per a finalitzar l'analisi exploratòria generaré un dendograma de clústering jeràrquic per mostra i un dendoframa de clústering jeràrquic per metabòlit.

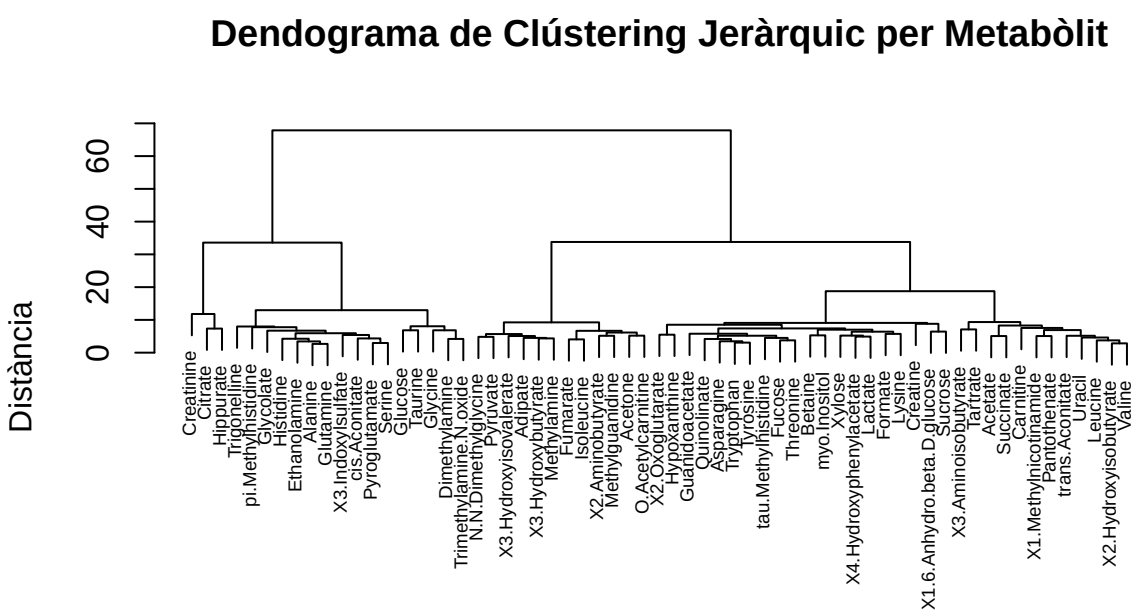
El Dendograma de clústering jeràrquic per mostra agrupa les mostres dels pacients en funció del ser perfil de metabòlits. Mentre estava fent l'estudi exploratori de les dades, vaig adonar-me que era més interessant calcular la distància mitjançant una correlació inversa (1 - correlació) entre les mostres ja que així, visualment, les mostres amb perfils de metabòlits similars o amb alta correlació es trobarien més a prop del dendograma. Això es fa amb la finalitat d'identificar subgrups de pacients amb perfils metabòlics similars.

```
#Així doncs, si existís una diferència marcada i clara en el dendograma, es podria mirar si els grups c


dades_metabolits_logaritmiques <- log2(assay(se) + 1)
dades_metaboliques_mod <- scale(dades_metabolits_logaritmiques)


#genero el dendograma de clústerin jeràrquic per mostra

mostra_dendograma <- as.dist(1 - cor(dades_metaboliques_mod))
mostra_jerarquica <- hclust(mostra_dendograma, method = "ward.D2")

#fem la representació visual del dendograma de clústering jerarquic per mostra
plot(mostra_jerarquica, main = "Dendograma de Clústering jeràrquic per mostra",
     xlab = "Mostres (Patient.ID)", ylab = "Distància", cex = 0.5)
```



Aquest dendograma per mostra el que fa és agrupar les diferents mostres de pacients en funció dels seus perfils metabòlics. Cada branca és una mostra i la seva proximitat indica la similitud en les seves concentracions de metabòlits. Aquest dendograma il · lustra alguns clústers ben definits, com, per exemple, el clúster central conformat per les mostres PIF_115 i PIF_100, suggerint que tenen perfils metabolòmics similars

```
#Ara genero el dendograma de clústering Jeràrquic per metabòlit. Aquest dendograma per metabòlit agrupa

mostra_dendograma_metabolit <- dist(dades_metaboliques_mod, method = "euclidean")
mostra_2 <- hclust(mostra_dendograma_metabolit, method = "ward.D2")

plot(mostra_2, main = "Dendograma de Clústering Jeràrquic per Metabòlit",
     xlab = "Metabòlits", ylab = "Distància", cex = 0.6)
```



Aquest dendograma proporciona informació sobre com els metabòlits s'agrupen en funció dels seus patrons de variació en les mostres analitzades. Pot ser útil per a identificar visualment relacions o associacions entre metabòlits És a dir, aquells metabòlits que es troben al mateix clúster tindran perfils de variacions similars. Com a exemple, podem veure un clúster ben diferenciat a la banda esquerra. Aquest clúster està conformat per la creatinina, hipoxantina, histidina, glicolat i l'etanolamina.

Discussió i limitacions i conclusions de l'estudi

A l'informe present s'ha realitzat una anàlisi inicial exploratòria del dataset human_cachexia el qual ha permès el seu objectiu principal, explorar les diferències metabolòmiques entre pacients oncològics normals vs pacients oncològics caquèxics mitjançant l'ús del contenidor SummarizedExperiment el qual ha facilitat aquesta tasca on la manipulació i organització de dades és constant.

No obstant això, la present anàlisi exploratòria presenta algunes limitacions, ja que la mostra de l'estudi té una mida relativament petita (77 pacients) i és difícil poder arribar a conclusions fermes. L'anàlisi exploratòria, tot hi que permet esclarir alguns aspectes metabolòmics inicials, no permet treure conclusions definitives o concluents (tot i que aquest anàlisi no està dirigit cap a això). Per a poder obtenir unes conclusions concloents i fermes seria necessari aplicar diferents proves estadístiques adaptades al nostre context. Una altre aspecte a tenir en consideració és que, per a poder visualitzar les dades de concentracions de metabòlits de forma còmoda a un histograma, va ser necessari dur a terme una transformació logarítmica. No obstant això, aquesta troballa indica la presència de valors extrems que pot suggerir la necessitat de dur a terme tècniques de normalització o preprocessament de dades. D'igual forma, encara que els dendogrames aportin informació interessant, haurien de fer-se estudis amb pacients amb la finalitat de confirmar o no si els clústers generats tenen correspondència amb diferències clíniques estadísticament significatives.

Així doncs, en conclusió, aquesta tasca ha permès l'aplicació de mètodes d'exploració de dades a un dataset metabolòmic i s'han vist potencials diferències als perfils de metabòlits dels pacients oncològics. Així doncs,

l'organització de la informació en un contenidor SummarizedExperiment i les visualitzacions com el PCA han facilitat una primera inspecció visual que pot arribar a apuntar cap a algunes tendències que hauran de ser confirmades en estudis posteriors. Finalment, aquesta pràctica estableix els fonaments per a futur anàlisis bioinformàtics més complexos i estadísticament més demandants.

link al repositori (Josep Rocaspana Codana)

https://github.com/Joseprrcc/Rocaspana-Codana-Josep-PAC1

Així doncs, arribo al final de l'analisi exploratori de les dades del dataset. he executat la comanda save(se, file = "se_contenidor_josep_rocaspana_codana.rda") a la consola per a obtenir l'objecte contenidor en format binari.