

Prova de d'avaluación continua 1

Bianca Teresa Castro Criado

30/03/2025

Índice

0.1	Resum	1
0.2	Objectius	1
0.3	Mètodes	1
0.4	Resultats	10
0.5	Discussió	11
0.6	Referències	11

0.1 Resum

Aquest informe presenta una anàlisi de dades de fosfoproteòmica de dos tipus de tumors (MSS I PD), el qual es centra en la seva organització, exploració i interpretació. Primer, s'ha seleccionat un conjunt de dades i s'ha estructurat en un objecte SummarizedExperiment, facilitant-ne la gestió i l'anàlisi. A continuació, s'han aplicat tècniques d'anàlisi exploratòria, incloent la normalització i la visualització mitjançant PCA, per identificar patrons i variabilitat en les dades. Els resultats obtinguts proporcionen informació sobre la distribució i característiques de les mostres. S'ha conclòs que els pèptids son deferents pels dos tipus de tumors.

0.2 Objectius

L'objectiu principal d'aquest treball és realitzar una anàlisi de les dades mitjançant eines bioinformàtiques i estadístiques, amb la finalitat d'obtenir coneixements rellevants sobre el conjunt de dades proporcionat. Concretament, es busca:

- Explorar les dades i identificar patrons significatius.
- Aplicar tècniques estadístiques per validar resultats.
- Integrar codi i anàlisi en un document clar i estructurat.

0.3 Mètodes

Les dades escollides són aquelles recollides a l'arxiu *TIO2+PTYR-human-MSS+MSIvsPD* del repositori de GitHub proporcionat. Les dades provenen de un experiment de fosfoproteòmica que ha analitzat (3 + 3) models PDX de dos subtipus diferents de tumors mitjançant mostres enriquides en fosfopèptids. L'anàlisi es

va dur a terme mitjançant LC-MS, amb 2 rèpliques tècniques per mostra. El conjunt de resultats consisteix en abundàncies normalitzades dels senyals de MS per a aproximadament 1400 fosfopèptids.

Tant `ExpressionSet` com `SummarizedExperiment` són classes emprades per gestionar dades òmiques, i tot i compartir algunes similituds, presenten diferències importants en flexibilitat i aplicació. `ExpressionSet`, que forma part del paquet `Biobase` i és més antic, conté tres components principals: `exprs()` per a la matriu d'expressió (valors de senyal gènic, proteic, RNA, etc), `pData()` per a les metadades de les mostres i `fData()` per a les metadades de la unitat biològica/anotacions. És adequat per a dades d'expressió tradicionals (microarrays o RNA-seq senzills), però és limitat per a dades complexes, ja que no permet gestionar múltiples matrius ni nivells de dades. En canvi, `SummarizedExperiment`, que és més recent que l'`ExpressionSet`, ofereix una estructura més flexible amb `assays()` per emmagatzemar una o més matrius d'expressió, `colData()` per a metadades de mostres, `rowData()` per a metadades de característiques i `metadata()` per informació addicional. Això el fa ideal per a dades obtingudes per metodologies més complexes, ja que permet gestionar múltiples tipus de dades.

A continuació es detalla el procés de creació d'un `SummarizedExperiment`:

```
library(readxl)
library(SummarizedExperiment)

## Loading required package: MatrixGenerics

## Loading required package: matrixStats

##
## Attaching package: 'MatrixGenerics'

## The following objects are masked from 'package:matrixStats':
##
##   colAlls, colAnyNAs, colAnys, colAvgPerRowSet, 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, rowAvgPerColSet,
##   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

## Loading required package: GenomicRanges

## Loading required package: stats4

## Loading required package: BiocGenerics

##
## Attaching package: 'BiocGenerics'
```

```

## 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, saveRDS, setdiff,
##   table, tapply, union, unique, unsplit, which.max, which.min

## Loading required package: S4Vectors

##
## Attaching package: 'S4Vectors'

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

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

## Loading required package: IRanges

## Loading required package: GenomeInfoDb

## Loading required package: Biobase

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

##
## Attaching package: 'Biobase'

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

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

```

```
# Primer s'importen les dades i s'assignen a una variable
dades <- "/Users/biancacaastrocriado/Desktop/metaboData/Datasets/2018-Phosphoproteomics/TI02+PTYR-human-1"
```

```
# Després, com a l'arxiu hi ha dos fulls, s'assigna al primer d'ells la ruta a l'arxiu
# excel amb les dades de la matriu i anotacions sobre els fosfopèptids.
# A la variable dades_mostres s'assignen les dades relatives a les mostres.
```

```
primer_full <- read_excel(dades, sheet = "originalData")
metadata_mostres <- read_excel(dades, sheet = "targets")
```

```
## New names:
## * 'Sample' -> 'Sample...1'
## * 'Sample' -> 'Sample...2'
```

```
# Llavors es crea primer el dataframe que es passarà a la funció SummarizedExperiment
# per crear l'objecte i que contrindrà la ColData que, com s'ha mencionat anteriorment,
# emmagatzema la informació relativa a les mostres, que es la informació que es troba
# a les columnes i que ve determinada per la persona investigadora (son les metadades
# de la mostra).
```

```
print(metadata_mostres)
```

```
## # A tibble: 12 x 4
##   Sample...1 Sample...2 Individual Phenotype
##   <chr>      <chr>      <dbl> <chr>
## 1 M1_1      M1              1 MSS
## 2 M1_2      M1              1 MSS
## 3 M5_1      M5              2 MSS
## 4 M5_2      M5              2 MSS
## 5 T49_1     T49              3 MSS
## 6 T49_2     T49              3 MSS
## 7 M42_1     M42              4 PD
## 8 M42_2     M42              4 PD
## 9 M43_1     M43              5 PD
## 10 M43_2    M43              5 PD
## 11 M64_1     M64              6 PD
## 12 M64_2    M64              6 PD
```

```
# Com es pot observar, la colData conté dades sobre les mostres per duplicat.
# On la primera columna conté el nom de la mostra que hauria de coincidir amb el nom
# de les columnes de l'Assay (matriu de dades). Si s'exploren els noms de les columnes
# que representen les mostres (més les anotacions):
```

```
colnames(primer_full)
```

```
## [1] "SequenceModifications" "Accession" "Description"
## [4] "Score" "M1_1_MSS" "M1_2_MSS"
## [7] "M5_1_MSS" "M5_2_MSS" "T49_1_MSS"
## [10] "T49_2_MSS" "M42_1_PD" "M42_2_PD"
## [13] "M43_1_PD" "M43_2_PD" "M64_1_PD"
## [16] "M64_2_PD" "CLASS" "PHOSPHO"
```

```

# Es pot comprovar que no coincideixen. Per tant, es canvia el format de la columna
# per que coincideixi.
metadata_mostres$`Sample...1` <-
  paste0(metadata_mostres$`Sample...1`, "_", metadata_mostres$Phenotype)
colnames(metadata_mostres)[colnames(metadata_mostres) == "Sample...2"] <- "sample"

# S'assigna com a nom de les mostres l'informació continguda a Sample...1
metadata_mostres <- as.data.frame(metadata_mostres)
rownames(metadata_mostres) <- metadata_mostres$`Sample...1`
metadata_mostres$`Sample...1` <- NULL

# Atès que l'experiment és de fosfoproteòmica la unitat biològica de cada fila són
# fosfopèptids, les files s'han d'identificar per un identificador únic per a cada
# fosfopèptid. Respecte a les columnes del dataset, cap de les variables sembla adequada
# com a identificador. Això és degut a que, per exemple, SequenceModifications conté la
# seqüència del pèptid juntament amb les seves modificacions post-traduccionals, informació
# que seria més adequada per la secció rowData. La variable description tampoc seria una
# bona opció perquè moltes files tenen la mateixa descripció, ja que diversos fosfopèptids
# poden provenir de la mateixa proteïna. De la mateixa manera, accession es el número
# identificatiu a Uniprot de la proteïna que pot patir una modificació post-traduccional a
# un dels seus pèptids, pel que tampoc seria adequada. Per tant, primer es crea el
# dataframe de rowData (que inclou anotacions sobre el pèptid en qüestió, que seria
# el metadata dels pèptids).

# Primer s'ha de crear un vector que contingui els identificadors dels pèptids.
# Aquest vector serà tan llarg com pèptids hi hagi.

peptide_names <- paste0("peptide ", seq_len(nrow(primer_full)))

# Després, es crea el dataframe que contindrà l'assay data, que seran les dades
# numèriques de l'abundància normalitzada de la senyal MS dels pèptids.
# També s'assigna com a nom de les files l'identificador dels pèptids.
assay_data <- data.frame(primer_full[5:16], row.names = peptide_names)

# Se segueix el mateix procediment per crear la secció de les anotacions,
# on s'emmagatzemarà la informació de SequenceModification, el accession number,
# description, Score (el que ha de ser una mètrica de qualitat d'identificació
# del fosfopèptid a l'espectrometria de masses), la classe i l'aminoàcid on succeeix
# la fosforilació. A tots aquests també se'ls assigna un identificador del pèptid.
# També s'assigna a les files el nom que es troba a la variable peptide_names.

metadata_peptides <- data.frame(primer_full[c(1:4,17,18)], row.names = peptide_names)

# Es crea l'objecte SummarizedExperiment
se <- SummarizedExperiment(
  assays = list(counts = assay_data),
  colData = metadata_mostres,
  rowData = metadata_peptides
)

se

```

```

## class: SummarizedExperiment
## dim: 1438 12

```

```
## metadata(0):
## assays(1): counts
## rownames(1438): peptide 1 peptide 2 ... peptide 1437 peptide 1438
## rowData names(6): SequenceModifications Accession ... CLASS PHOSPHO
## colnames(12): M1_1_MSS M1_2_MSS ... M64_1_PD M64_2_PD
## colData names(3): sample Individual Phenotype
```

```
# S'explora l'objecte SummarizedExperiment
# Resum de l'objecte
print(se)
```

```
## class: SummarizedExperiment
## dim: 1438 12
## metadata(0):
## assays(1): counts
## rownames(1438): peptide 1 peptide 2 ... peptide 1437 peptide 1438
## rowData names(6): SequenceModifications Accession ... CLASS PHOSPHO
## colnames(12): M1_1_MSS M1_2_MSS ... M64_1_PD M64_2_PD
## colData names(3): sample Individual Phenotype
```

```
# Matriu de dades
head(assay(se))
```

```
##           M1_1_MSS  M1_2_MSS  M5_1_MSS  M5_2_MSS  T49_1_MSS  T49_2_MSS
## peptide 1      24.29438  44475.964      0.000   6269.141   1135.8169   21933.90
## peptide 2       0.00000  43138.904     2102.056  50355.051    248.9275    3239.16
## peptide 3    3412.60332 172143.040    77323.019 307637.429   98442.2773   192982.37
## peptide 4  220431.17880 145656.887   104287.815   75887.365   773377.4981   481165.54
## peptide 5   18254.77813   8529.755   35955.901   44102.316   57145.1682    34638.01
## peptide 6  644513.31840 261938.025  187023.484 124867.715  4487443.6920  2572575.27
##           M42_1_PD  M42_2_PD  M43_1_PD  M43_2_PD  M64_1_PD
## peptide 1       0.000      0.00    772.9056   2136.746   1820.724
## peptide 2     1315.904      0.00      0.0000      0.000      0.000
## peptide 3     24851.344   16547.95    5565.2821      0.000   3264.563
## peptide 4  1027196.292 1163747.38  4080239.1820  4885818.113  3093786.793
## peptide 5     21231.256   49499.70   666107.0448   379313.615   255792.117
## peptide 6   535809.187  434645.89   91361.8781   65997.913   243250.439
##           M64_2_PD
## peptide 1     1727.9098
## peptide 2      892.3565
## peptide 3     5901.9577
## peptide 4  2759104.5440
## peptide 5   579765.0018
## peptide 6   206632.6444
```

```
# Metadades de les anotacions
head(rowData(se))
```

```
## DataFrame with 6 rows and 6 columns
##           SequenceModifications  Accession  Description  Score
##           <character> <character>         <character> <numeric>
## peptide 1 LYPELSQYMGLSLNEEEIR[...  000560 Syntenin-1 OS=Homo s...    48.07
```

```
## peptide 2 VDKVIQAQTAFSANPANPAI..      000560 Syntenin-1 OS=Homo s..      67.05
## peptide 3 VIQAQTAFSANPANPAILSE..      000560 Syntenin-1 OS=Homo s..      77.71
## peptide 4 HADAEMTGYYVTR[6] Oxi..      015264 Mitogen-activated pr..      44.87
## peptide 5 HADAEMTGYYVTR[9] Pho..      015264 Mitogen-activated pr..      67.42
## peptide 6 STGPGASLGTGYDR[12] P..      015551 Claudin-3 OS=Homo sa..      63.69
##              CLASS      PHOSPHO
##      <character> <character>
## peptide 1          H          Y
## peptide 2          H          Y
## peptide 3          H          Y
## peptide 4          H          Y
## peptide 5          H          Y
## peptide 6          H          Y
```

```
# Metadades de les mostres
colData(se)
```

```
## DataFrame with 12 rows and 3 columns
##              sample Individual  Phenotype
##      <character> <numeric> <character>
## M1_1_MSS          M1          1          MSS
## M1_2_MSS          M1          1          MSS
## M5_1_MSS          M5          2          MSS
## M5_2_MSS          M5          2          MSS
## T49_1_MSS         T49          3          MSS
## ...              ...          ...          ...
## M42_2_PD          M42          4          PD
## M43_1_PD          M43          5          PD
## M43_2_PD          M43          5          PD
## M64_1_PD          M64          6          PD
## M64_2_PD          M64          6          PD
```

```
#Es guarden les dades en els formats especificats a l'enunciat:
```

```
#SummarizedExperiment:
```

```
save(se, file = "SummarizedExperiment_phospho.Rda")
```

```
#Per carregar-ho posteriorment en un altre arxiu: load("SummarizedExperiment_phospho.Rda")
```

```
# Dades en format .txt amb separador tabulador. Es crea un .txt per acada tipus de dades
```

```
write.table(assay(se), file = "assay_data.txt", sep = "\t", row.names = TRUE, quote = FALSE)
```

```
write.table(rowData(se), file = "peptide_metadata.txt", sep = "\t", row.names = TRUE, quote = FALSE)
```

```
write.table(colData(se), file = "samples_metadata.txt", sep = "\t", row.names = TRUE, quote = FALSE)
```

```
# Es crea l'arxiu markdown amb la descripció de les dades
```

```
cat("# Metadades de l'experiment\n
```

```
Aquest arxiu acompanya les dades exportades.\n
```

```
## arxius:\n
```

```
- assay_data.txt: Matriu d'abundància dels pèptids amb les dades quantitatives.\n
```

```
- peptide_metadata.txt: Metadades relatives a les mostres, que inclou: el nom de la mostra, l'individu i el fenotip\n
```

```
- samples_metadata.txt: Metadades dels pèptids, que inclouen el accesion number de Uniprot, la seqüència dels pèptids, la descripció de la proteïna on es troba aquest pèptid, l'Score, la classe y l'aminoàcid on succeeix la fosforilació.\n
```

```
", file = "metadades.md")
```

Anàlisi estadístic:

```
summary(assay(se))
```

```
##      M1_1_MSS      M1_2_MSS      M5_1_MSS      M5_2_MSS
## Min.      :      0   Min.      :      0   Min.      :      0   Min.      :      0
## 1st Qu.:   5653   1st Qu.:   5497   1st Qu.:   2573   1st Qu.:   3273
## Median :   30682  Median :   26980  Median :   20801  Median :   26241
## Mean    :  229841  Mean    :  253151  Mean    :  232967  Mean    :  261067
## 3rd Qu.:  117373  3rd Qu.:  113004  3rd Qu.:  113958  3rd Qu.:  130132
## Max.    :16719906  Max.    :43928481  Max.    :15135169  Max.    :19631820
##      T49_1_MSS      T49_2_MSS      M42_1_PD      M42_2_PD
## Min.      :      0   Min.      :      0   Min.      :      0   Min.      :      0
## 1st Qu.:   9306   1st Qu.:   8611   1st Qu.:   5341   1st Qu.:   4216
## Median :   55641  Median :   46110  Median :   36854  Median :   30533
## Mean    :   542449  Mean    :   462616  Mean    :   388424  Mean    :   333587
## 3rd Qu.:  223103  3rd Qu.:  189141  3rd Qu.:  180252  3rd Qu.:  152088
## Max.    :49218872  Max.    :29240206  Max.    :48177680  Max.    :42558111
##      M43_1_PD      M43_2_PD      M64_1_PD      M64_2_PD
## Min.      :      0   Min.      :      0   Min.      :      0   Min.      :      0
## 1st Qu.:   19641  1st Qu.:   17299  1st Qu.:   11038  1st Qu.:   8660
## Median :   67945  Median :   59607  Median :   52249  Median :   47330
## Mean    :   349020  Mean    :   358822  Mean    :   470655  Mean    :   484712
## 3rd Qu.:  205471  3rd Qu.:  201924  3rd Qu.:  209896  3rd Qu.:  206036
## Max.    :35049402  Max.    :63082982  Max.    :71750330  Max.    :88912734
```

```
library(ggplot2)
library(tidyr)
```

```
##
## Attaching package: 'tidyr'
```

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

```
# S'aplica una transformació logarítmica, ja que Les dades tenen una distribució
# molt asimètrica, amb valors mínims de 0 i màxims que arriben a desenes de milions.
# Això fa que la distribució estigui fortament esbiaixada a la dreta (molts valors petits
# i uns pocs valors molt grans). L'aplicació del logaritme ajuda a reduir aquest
# efecte i a fer que la distribució sigui més propera a una normal
dades <- as.data.frame(log10(assay(se) + 1))

# Es trasposa la matriu perquè cada columna sigui una fila
dades <- t(dades)
dades <- as.data.frame(dades)

# S'agrega la informació de les mostres desde colData
dades$Mostra <- colData(se)$sample # Ajusta "sample" según el nombre correcto en colData(se)

# La següent línia transforma dades de format ample a llarg, de manera que converteix
# les columnes de pèptids en files. Manté la variable Mostra fixa, crea una columna "Feature"
```

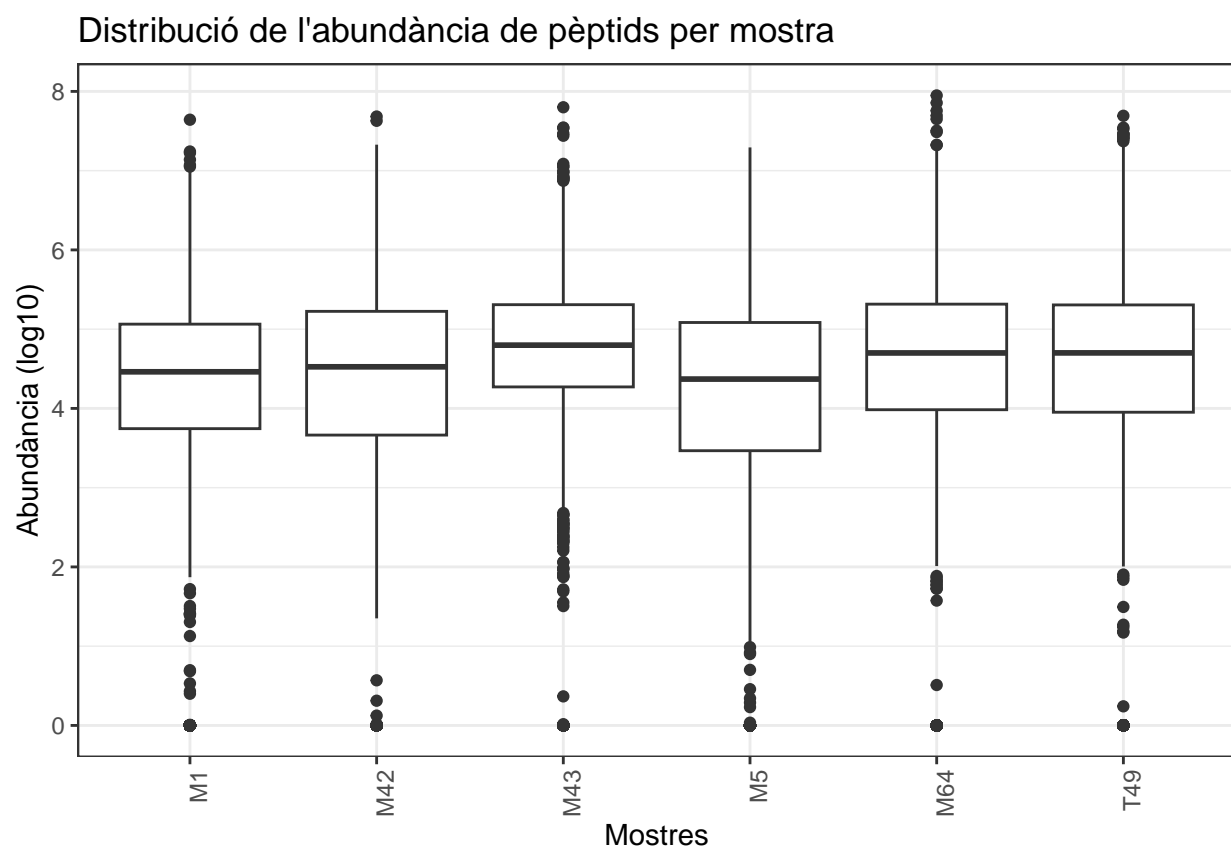


```

# amb els noms dels pèptids i una columna "Abundància" amb els valors corresponents.
# Així, cada fila representa una combinació de mostra, pèptid i abundància, el que facilita
# la creació del boxplot.
dades_long <- pivot_longer(dades, cols = -Mostra, names_to = "Feature", values_to = "Abundància")

ggplot(dades_long, aes(x = Mostra, y = Abundància)) +
  geom_boxplot() +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  labs(title = "Distribució de l'abundància de pèptids per mostra",
       x = "Mostres",
       y = "Abundància (log10)")

```



Tal i com es comenta al arxiu que descriu les dades de l'arxiu original, l'objectiu de les dades és buscar fosfopèptids que permetin la diferenciació de dos grups tumorals. Per tal de fer això es pot recórrer a tècniques com el PCA.

```

# Primer ha d'extreure's la matriu d'abundàncies
abundances <- assay(se)
matriu <- as.matrix(log10(abundances + 1))

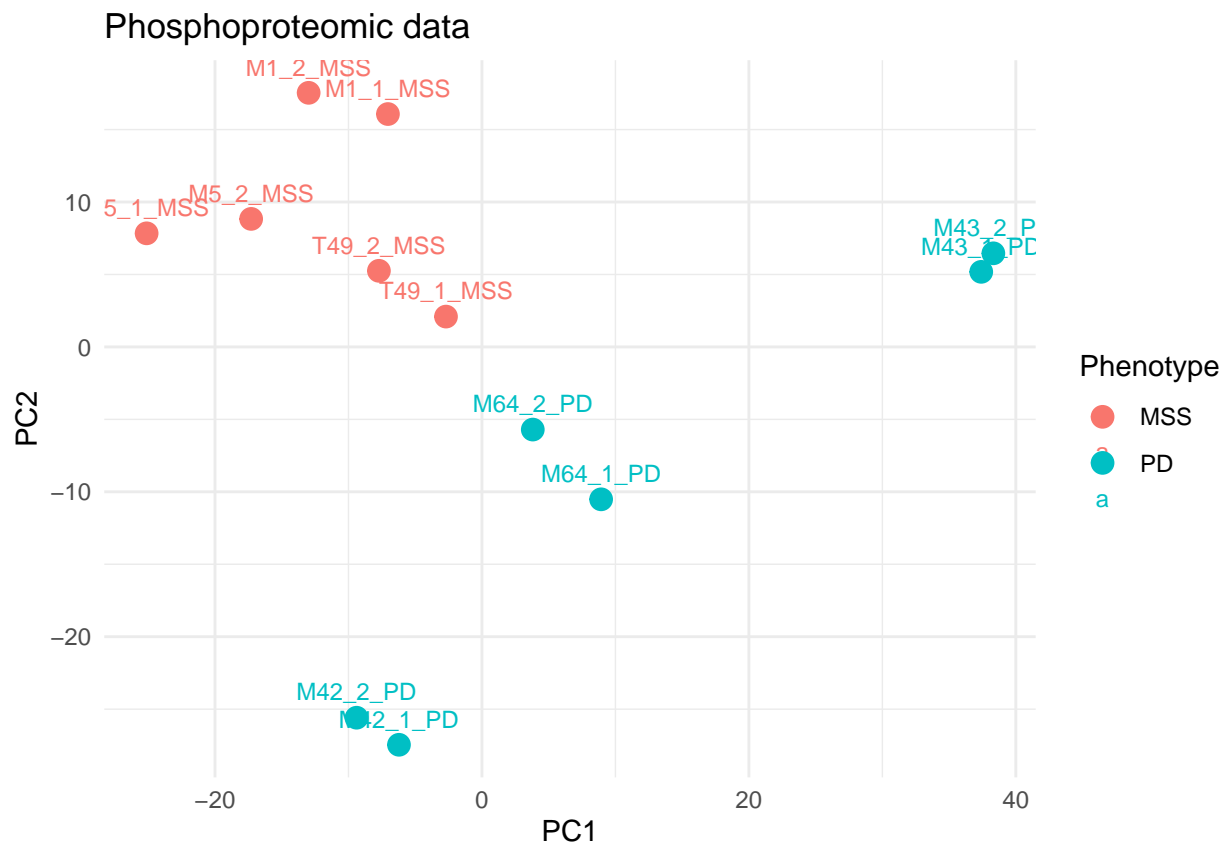
# Es calcula el PCA
pca_res <- prcomp(t(matriu), scale. = FALSE)

# Es crea un dataframe amb les dades i s'hi afegeixen els noms de la mostra i el

```

```
# fenotip/tipus de tumor. Les metadates s'extreuen directament de l'objecte SummarizedExperiment
pca_data <- as.data.frame(pca_res$x)
pca_data$Sample <- colnames(abundances)
pca_data$Phenotype <- colData(se)$Phenotype

ggplot(pca_data, aes(x = PC1, y = PC2, color = Phenotype, label = Sample)) +
  geom_point(size = 3.5) +
  geom_text(vjust = -1, size = 3) +
  theme_minimal() +
  ggtitle("Phosphoproteomic data") +
  labs(x = "PC1", y = "PC2")
```



0.4 Resultats

L'anàlisi estadística descriptiva mostra una gran variabilitat en l'abundància, amb valors mínims de 0 i màxims que arriben als milions. Aquesta àmplia dispersió suggereix la presència d'alguns pèptids altament abundants juntament amb molts de baixa abundància. A més, la mitjana és significativament superior a la mediana en totes les mostres, indicant una distribució esbiaixada cap a la dreta amb valors extrems que influeixen en la mitjana. També s'observen diferències entre les mostres, amb les etiquetades com a PD mostrant en general valors mitjans i màxims més elevats en comparació amb les MSS, fet que podria reflectir diferències biològiques o experimentals. A més, la presència de valors mínims de 0 en totes les mostres indica que alguns pèptids no han estat detectats en determinades condicions. Es fa una representació de les dades per captar aquesta variabilitat.

Confirmant el que s'ha dit abans, en el boxplot s'observa una gran variabilitat en l'abundància, amb valors

mínims propers a 0 i màxims molt elevats. L'existència de valors extrems indica la presència de pèptids altament abundants en algunes mostres. Les medianes són relativament similars entre les mostres.

El gràfic PCA mostra una clara separació entre els grups MSS i PD, el que indica que les seves signatures fosfoproteòmiques són diferents. Els punts que representen les mostres MSS (en vermell) es troben agrupats de manera més compacta, el que suggereix que aquest grup és més homogeni en termes de variació molecular. D'altra banda, les mostres del grup PD (en blau) es dispersen més al llarg dels eixos principals, cosa que indica una major heterogeneïtat dins d'aquest grup. Aquesta variabilitat en el grup PD podria estar relacionada amb patrons de fosforilació més versàtils dins d'aquest tipus de tumor, o bé amb una resposta diferencial a factors externs com ara el tractament o l'estat del tumor en relació a la fase clínica de la malaltia. La distribució dels punts també suggereix que la primera component principal és la que captura la major part de la variació entre els dos grups, mentre que la segona component podria estar reflectint diferències individuals dins de cada grup.

0.5 Discussió

Els resultats obtinguts a partir de l'anàlisi de fosfoproteòmica permeten identificar patrons diferencials en les mostres analitzades. Aquests patrons en un context biològic indiquen possibles vies de regulació diferencial sorgides de la diferència entre tumors en la fosforilació dels pèptids. L'anàlisi està subjecta a diverses limitacions, incloent la qualitat i el pretractament de les dades. La presència de valors extrems poden haver influït en els resultats. A més, la selecció del mètode de normalització pot haver afectat la interpretació de les diferències observades entre grups experimentals. Per millorar la robustesa de l'anàlisi, es podria augmentar la mida de la mostra i aplicar tècniques avançades de machine learning per detectar patrons més complexos. A més, es podrien integrar altres fonts de dades òmiques per proporcionar una visió més completa dels mecanismes biològics subjacents.

0.6 Referències

Repositori de Github