Anàlisi de Components Principals

Paula Solé Vallés

2024-12-03

Abans de realitzar la integració, realitzem un Anàlisis de Components Principals per assegurar-nos que els diferents grups en que es classifiquen les nostres dades es separen. Per a fer-ho utilitzem els 3 blocs de dades que tenim: - Transcriptòmica - Metabolòmica - Paràmetres fisiològics

Obtenció de les dades

```
# Set working directory

setwd('/Volumes/ftp/Paula Sole')

# Upload files
transcriptomics <- read_excel("FPKM_tomate_inundacion_all.xlsx")
metabolites_PA <-
read_excel("Summary_metabolites_hormones_2019_luk&not_PA.xlsx",sheet =
"Full2")

## New names:
## • `` -> `...1`
## • `LUKCT` -> `LUKCT...8`
## • `LUKCT` -> `LUKCT...9`
## • `LUKCT` -> `LUKCT...10`
## • `LUKCT` -> `LUKCT...11`
## • `LUKST` -> `LUKST...12`
## • `LUKST` -> `LUKST...13`
## • `LUKST` -> `LUKST...14`
## • `LUKST` -> `LUKST...15`
## • `PVALUE` -> `PVALUE...16`
## • `FC` -> `FC...17`
## • `` -> `...18`
## • `notCT` -> `notCT...19`
## • `notCT` -> `notCT...20`
## • `notCT` -> `notCT...21`
## • `notCT` -> `notCT...22`
## • `notST` -> `notST...23`
## • `notST` -> `notST...24`
## • `notST` -> `notST...25`
## • `notST` -> `notST...26`
## • `PVALUE` -> `PVALUE...27`
## • `FC` -> `FC...28`
```

```r
metabolites_AR <-
read_excel("Summary_metabolites_hormones_2019_luk&not_AR.xlsx",sheet =
"duplicat")

## New names:
## • `` -> `...1`
## • `LUKCT` -> `LUKCT...8`
## • `LUKCT` -> `LUKCT...9`
## • `LUKCT` -> `LUKCT...10`
## • `LUKCT` -> `LUKCT...11`
## • `LUKST` -> `LUKST...12`
## • `LUKST` -> `LUKST...13`
## • `LUKST` -> `LUKST...14`
## • `LUKST` -> `LUKST...15`
## • `PVALUE` -> `PVALUE...16`
## • `FC` -> `FC...17`
## • `` -> `...18`
## • `notCT` -> `notCT...19`
## • `notCT` -> `notCT...20`
## • `notCT` -> `notCT...21`
## • `notCT` -> `notCT...22`
## • `notST` -> `notST...23`
## • `notST` -> `notST...24`
## • `notST` -> `notST...25`
## • `notST` -> `notST...26`
## • `PVALUE` -> `PVALUE...27`
## • `FC` -> `FC...28`

physiological <-
read_excel("Summary_Physiological_parameters_lukullus_notabilis_PA_2019.xlsx"
, sheet = "Full3")

## New names:
## • `` -> `...1`
```

Separem les dades segons els grups de mostres:

```r
# Set the row names for the dataframe
transcriptomics <- as.data.frame(transcriptomics)
rownames(transcriptomics) <- transcriptomics$Locus
# Remove the first column as it's now used as row names
transcriptomics <- transcriptomics[-1]

# PA dataset
transcriptomics_PA <- transcriptomics[, grepl("PA",
colnames(transcriptomics))]
transcriptomics_PA <- t(transcriptomics_PA)

# AR dataset
transcriptomics_AR <- transcriptomics[, grepl("AR",
colnames(transcriptomics))]
```

```r
transcriptomics_AR <- t(transcriptomics_AR)

# Luk dataset
transcriptomics_Luk <- transcriptomics[, grepl("Lukullus",
colnames(transcriptomics))]
transcriptomics_Luk <- t(transcriptomics_Luk)

# Not dataset
transcriptomics_Not <- transcriptomics[, grepl("Lukullus",
colnames(transcriptomics))]
transcriptomics_Not <- t(transcriptomics_Not)

head(metabolites_PA)
```

```
## # A tibble: 6 × 28
##   ...1      annotation    mz    rt `rt[min]` isotopes adduct LUKCT...8
LUKCT...9
##   <chr>     <chr>      <dbl> <dbl>     <dbl> <chr>    <chr>  <chr>
<chr>
## 1 <NA>      <NA>         NA    NA      NA    <NA>     <NA>   SCIC_VD_…
SCIC_VD_…
## 2 metLCpos… Unknown 4…  419.  239.      3.98 <NA>     [M+H]… 88.83246…
87.01376…
## 3 metLCpos… Tryptophan  188.  242.      4.03 [14][M]+ [M+K]… 4.179047…
4.866285…
## 4 metLCpos… Unknown 3…  329.  253.      4.21 [59][M]+ <NA>   2.201349…
2.276503…
## 5 metLCpos… feruloyl …  265.  285.      4.75 <NA>     <NA>   0.509605…
0.720923…
## 6 metLCpos… Chlorogen…  355.  294.      4.90 [67][M]+ [M+H]… 3.024634…
10.74253…
## # ℹ 19 more variables: LUKCT...10 <chr>, LUKCT...11 <chr>, LUKST...12
<chr>,
## #   LUKST...13 <chr>, LUKST...14 <chr>, LUKST...15 <chr>, PVALUE...16
<dbl>,
## #   FC...17 <chr>, ...18 <lgl>, notCT...19 <chr>, notCT...20 <chr>,
## #   notCT...21 <chr>, notCT...22 <chr>, notST...23 <chr>, notST...24
<chr>,
## #   notST...25 <chr>, notST...26 <chr>, PVALUE...27 <dbl>, FC...28 <chr>
```

```r
tail(metabolites_PA)
```

```
## # A tibble: 6 × 28
##   ...1  annotation        mz    rt `rt[min]` isotopes adduct LUKCT...8
LUKCT...9
##   <chr> <chr>          <dbl> <dbl>     <dbl> <chr>    <chr>  <chr>
<chr>
## 1 <NA>  <NA>              NA    NA        NA <NA>     <NA>   <NA>
<NA>
## 2 <NA>  Annotation        NA    NA        NA <NA>     <NA>   LUKCT
```

```
LUKCT
## 3 <NA>  Abscisic acid     NA    NA        NA <NA>     <NA>   131.4762…
179.7418…
## 4 <NA>  Phaseic acid      NA    NA        NA <NA>     <NA>   33.24196…
44.14035…
## 5 <NA>  Jasmonic acid     NA    NA        NA <NA>     <NA>   4.903905…
3.671501…
## 6 <NA>  Jasmonoyl iso…    NA    NA        NA <NA>     <NA>   nd        nd
## # ℹ 19 more variables: LUKCT...10 <chr>, LUKCT...11 <chr>, LUKST...12
<chr>,
## #   LUKST...13 <chr>, LUKST...14 <chr>, LUKST...15 <chr>, PVALUE...16
<dbl>,
## #   FC...17 <chr>, ...18 <lgl>, notCT...19 <chr>, notCT...20 <chr>,
## #   notCT...21 <chr>, notCT...22 <chr>, notST...23 <chr>, notST...24
<chr>,
## #   notST...25 <chr>, notST...26 <chr>, PVALUE...27 <dbl>, FC...28 <chr>
```

```r
# Delete the first row (it does not have important information)
metabolites_PA <- metabolites_PA[-c(1),]

# Delete the last rows as the information they contain is not relevant for
our analysis
metabolites_PA <- metabolites_PA[-c(117:122),]

# Delete the columns that do not have relevant information
names(metabolites_PA)
```

```
##  [1] "...1"        "annotation"  "mz"          "rt"          "rt[min]"
##  [6] "isotopes"    "adduct"      "LUKCT...8"   "LUKCT...9"   "LUKCT...10"
## [11] "LUKCT...11"  "LUKST...12"  "LUKST...13"  "LUKST...14"  "LUKST...15"
## [16] "PVALUE...16" "FC...17"     "...18"       "notCT...19"  "notCT...20"
## [21] "notCT...21"  "notCT...22"  "notST...23"  "notST...24"  "notST...25"
## [26] "notST...26"  "PVALUE...27" "FC...28"
```

```r
metabolites_PA <- metabolites_PA[-c(3,4,5,6,7,17,18,28)]
names(metabolites_PA)
```

```
##  [1] "...1"        "annotation"  "LUKCT...8"   "LUKCT...9"   "LUKCT...10"
##  [6] "LUKCT...11"  "LUKST...12"  "LUKST...13"  "LUKST...14"  "LUKST...15"
## [11] "PVALUE...16" "notCT...19"  "notCT...20"  "notCT...21"  "notCT...22"
## [16] "notST...23"  "notST...24"  "notST...25"  "notST...26"  "PVALUE...27"
```

```r
# Convert the tibble to a data frame
metabolites_PA <- as.data.frame(metabolites_PA)
# Set the row names for the dataframe
rownames(metabolites_PA) <- metabolites_PA$...1
# Remove the first column if it’s now used as row names
metabolites_PA <- metabolites_PA[-1]
# Transpose the data: samples in rows matebolits in columns
metabolites_PA <- t(metabolites_PA)
```

```r
# Select only the rows with metabolite information
# Delete the last sample to match number of samples in transcriptomics
dataset
rownames(metabolites_PA)
```

```
##  [1] "annotation"  "LUKCT...8"   "LUKCT...9"   "LUKCT...10"  "LUKCT...11"
##  [6] "LUKST...12"  "LUKST...13"  "LUKST...14"  "LUKST...15"  "PVALUE...16"
## [11] "notCT...19"  "notCT...20"  "notCT...21"  "notCT...22"  "notST...23"
## [16] "notST...24"  "notST...25"  "notST...26"  "PVALUE...27"
```

```r
metabolites_PA <- metabolites_PA[c(2,3,4,6,7,8,11,12,13,15,16,17),]
metabolites_PA <- as.data.frame(metabolites_PA)

head(metabolites_AR)
```

```
## # A tibble: 6 × 28
##   ...1      annotation    mz    rt `rt[min]` isotopes adduct LUKCT...8
LUKCT...9
##   <chr>     <chr>      <dbl> <dbl>     <dbl> <chr>    <chr>  <chr>
<chr>
## 1 <NA>      <NA>         NA    NA      NA    <NA>     <NA>   SCIC_VD_…
SCIC_VD_…
## 2 metLCpos… putative …  193.  240.      4.00 [13][M]+ [M+K+… 3.124284…
3.450654…
## 3 metLCpos… Tryptophan  205.  242.      4.03 <NA>     [M+K+… 0.556447…
0.480498…
## 4 metLCpos… feruloyl …  265.  285.      4.76 [27][M]+ <NA>   4.140352…
4.718082…
## 5 metLCpos… putative …  384.  295.      4.92 [91][M]+ [M+H]… 1.016770…
1.467846…
## 6 metLCpos… putative …  439.  298.      4.96 [113][M… [M+Na… 2.072486…
2.361271…
## # ℹ 19 more variables: LUKCT...10 <chr>, LUKCT...11 <chr>, LUKST...12
<chr>,
## #   LUKST...13 <chr>, LUKST...14 <chr>, LUKST...15 <chr>, PVALUE...16
<dbl>,
## #   FC...17 <chr>, ...18 <lgl>, notCT...19 <chr>, notCT...20 <chr>,
## #   notCT...21 <chr>, notCT...22 <chr>, notST...23 <chr>, notST...24
<chr>,
## #   notST...25 <chr>, notST...26 <chr>, PVALUE...27 <dbl>, FC...28 <chr>
```

```r
tail(metabolites_AR)
```

```
## # A tibble: 6 × 28
##   ...1  annotation        mz    rt `rt[min]` isotopes adduct LUKCT...8
LUKCT...9
##   <chr> <chr>          <dbl> <dbl>     <dbl> <chr>    <chr>  <chr>
<chr>
## 1 <NA>  <NA>              NA    NA        NA <NA>     <NA>   <NA>
<NA>
```

```
## 2 <NA>  Annotation        NA    NA        NA <NA>     <NA>   LUKCT
LUKCT
## 3 <NA>  Abscisic acid     NA    NA        NA <NA>     <NA>   54.98424…
40.54708…
## 4 <NA>  Phaseic acid      NA    NA        NA <NA>     <NA>   2.887671…
1.599270…
## 5 <NA>  Jasmonic acid     NA    NA        NA <NA>     <NA>   14.65547…
12.15145…
## 6 <NA>  Jasmonoyl iso…    NA    NA        NA <NA>     <NA>   11.68321…
9.919343…
## # ℹ 19 more variables: LUKCT...10 <chr>, LUKCT...11 <chr>, LUKST...12
<chr>,
## #   LUKST...13 <chr>, LUKST...14 <chr>, LUKST...15 <chr>, PVALUE...16
<dbl>,
## #   FC...17 <chr>, ...18 <lgl>, notCT...19 <chr>, notCT...20 <chr>,
## #   notCT...21 <chr>, notCT...22 <chr>, notST...23 <chr>, notST...24
<chr>,
## #   notST...25 <chr>, notST...26 <chr>, PVALUE...27 <dbl>, FC...28 <chr>
```

```r
# Delete the first row (it does not have important information)
metabolites_AR <- metabolites_AR[-c(1),]

# Delete the last rows as the information they contain is not relevant for
our analysis
metabolites_AR <- metabolites_AR[-c(142:147),]

# Delete the columns that do not have relevant information
names(metabolites_AR)
```

```
##  [1] "...1"        "annotation"  "mz"          "rt"          "rt[min]"
##  [6] "isotopes"    "adduct"      "LUKCT...8"   "LUKCT...9"   "LUKCT...10"
## [11] "LUKCT...11"  "LUKST...12"  "LUKST...13"  "LUKST...14"  "LUKST...15"
## [16] "PVALUE...16" "FC...17"     "...18"       "notCT...19"  "notCT...20"
## [21] "notCT...21"  "notCT...22"  "notST...23"  "notST...24"  "notST...25"
## [26] "notST...26"  "PVALUE...27" "FC...28"
```

```r
metabolites_AR <- metabolites_AR[-c(3,4,5,6,7,17,18,28)]
names(metabolites_AR)
```

```
##  [1] "...1"        "annotation"  "LUKCT...8"   "LUKCT...9"   "LUKCT...10"
##  [6] "LUKCT...11"  "LUKST...12"  "LUKST...13"  "LUKST...14"  "LUKST...15"
## [11] "PVALUE...16" "notCT...19"  "notCT...20"  "notCT...21"  "notCT...22"
## [16] "notST...23"  "notST...24"  "notST...25"  "notST...26"  "PVALUE...27"
```

```r
# Convert the tibble to a data frame
metabolites_AR <- as.data.frame(metabolites_AR)
# Set the row names for the dataframe
rownames(metabolites_AR) <- metabolites_AR$...1
# Remove the first column if it's now used as row names
metabolites_AR <- metabolites_AR[-1]
# Transpose the data: samples in rows matebolits in columns
```

```r
metabolites_AR <- t(metabolites_AR)

# Select only the rows with metabolite information
# Delete the last sample to match number of samples in transcriptomics
dataset
rownames(metabolites_AR)
```

```
##  [1] "annotation"  "LUKCT...8"   "LUKCT...9"   "LUKCT...10"  "LUKCT...11"
##  [6] "LUKST...12"  "LUKST...13"  "LUKST...14"  "LUKST...15"  "PVALUE...16"
## [11] "notCT...19"  "notCT...20"  "notCT...21"  "notCT...22"  "notST...23"
## [16] "notST...24"  "notST...25"  "notST...26"  "PVALUE...27"
```

```r
metabolites_AR <- metabolites_AR[c(2,3,4,6,7,8,11,12,13,15,16,17),]
metabolites_AR <- as.data.frame(metabolites_AR)

# Create the "Lukullus" dataset by selecting columns with "LUK" in their
names
metabolites_AR_Luk <- metabolites_AR[grepl("LUK", rownames(metabolites_AR)),]
metabolites_PA_Luk <- metabolites_PA[grepl("LUK", rownames(metabolites_PA)),]

common_columns <- intersect(colnames(metabolites_AR_Luk),
colnames(metabolites_PA_Luk))
metabolites_AR_Luk <- metabolites_AR_Luk[, common_columns]
metabolites_PA_Luk <- metabolites_PA_Luk[, common_columns]
metabolites_LUK <- rbind(metabolites_AR_Luk, metabolites_PA_Luk)



# Create the "notabilis" dataset by selecting columns whose names start with
"not"
metabolites_AR_Not <- metabolites_AR[grepl("not", rownames(metabolites_AR)),]
metabolites_PA_Not <- metabolites_PA[grepl("not", rownames(metabolites_PA)),]

common_columns <- intersect(colnames(metabolites_AR_Not),
colnames(metabolites_PA_Not))
metabolites_AR_Not <- metabolites_AR_Not[, common_columns]
metabolites_PA_Not <- metabolites_PA_Not[, common_columns]
metabolites_NOT <- rbind(metabolites_AR_Not, metabolites_PA_Not)

# Convert the tibble to a data frame
physiological <- as.data.frame(physiological)

# Set row names using the first column
rownames(physiological) <- physiological[[1]]

# Remove the first column as it is now used as row names
physiological <- physiological[-1]

# Create the "Lukullus" dataset by selecting columns with "LUK" in their
names
physiological_LUK <- physiological[grepl("LUK", rownames(physiological)), ]
```

```r
# Create the "Notabilis" dataset by selecting columns with "NOT" in their
names
physiological_NOT<- physiological[grepl("NOT", rownames(physiological)),]
```

PCA segons teixit

PA

Creem un dataframe que contingui les dades de PA tant de Lukullus com Notabillis i dels 3 tipus d'anàlisis. Les variables en columnes i les mostres en files.

```r
# Create a large datadrame with all the PA data
PA_dataframe <- cbind(physiological, transcriptomics_PA, metabolites_PA)
dim(PA_dataframe)

## [1]    12 29229
```

Modifiquem el dataframe de manera adequada per a realitzar el PCA.

```r
# Save group names
groups_PA <- row.names(PA_dataframe)
print(groups_PA)

##  [1] "LUK CT1/CT2"  "LUK CT3/CT4"  "LUK CT4/CT5"  "LUK ST1/ST2"  "LUK
ST3/ST4"
##  [6] "LUK ST9/ST10" "NOT CT1/CT2"  "NOT CT3/CT4"  "NOT CT5/CT6"  "NOT
ST1/ST2"
## [11] "NOT ST3/ST4"  "NOT ST9"

# Create uniform names for the samples
groups_PA <- gsub(".*(LUK|NOT).*(C|S).*", "\\1_\\2", groups_PA)
print(groups_PA)

##  [1] "LUK_C" "LUK_C" "LUK_C" "LUK_S" "LUK_S" "LUK_S" "NOT_C" "NOT_C"
"NOT_C"
## [10] "NOT_S" "NOT_S" "NOT_S"

# Convert the dataframe to numeric values
PA_dataframe <- apply(PA_dataframe, 2, as.numeric)
row.names(PA_dataframe) <- groups_PA

# Delete columns with NA
columns_NA <- colSums(is.na(PA_dataframe)) > 0
index_columns_NA <- which(columns_NA)
print(index_columns_NA)

## LWP
##   3
```

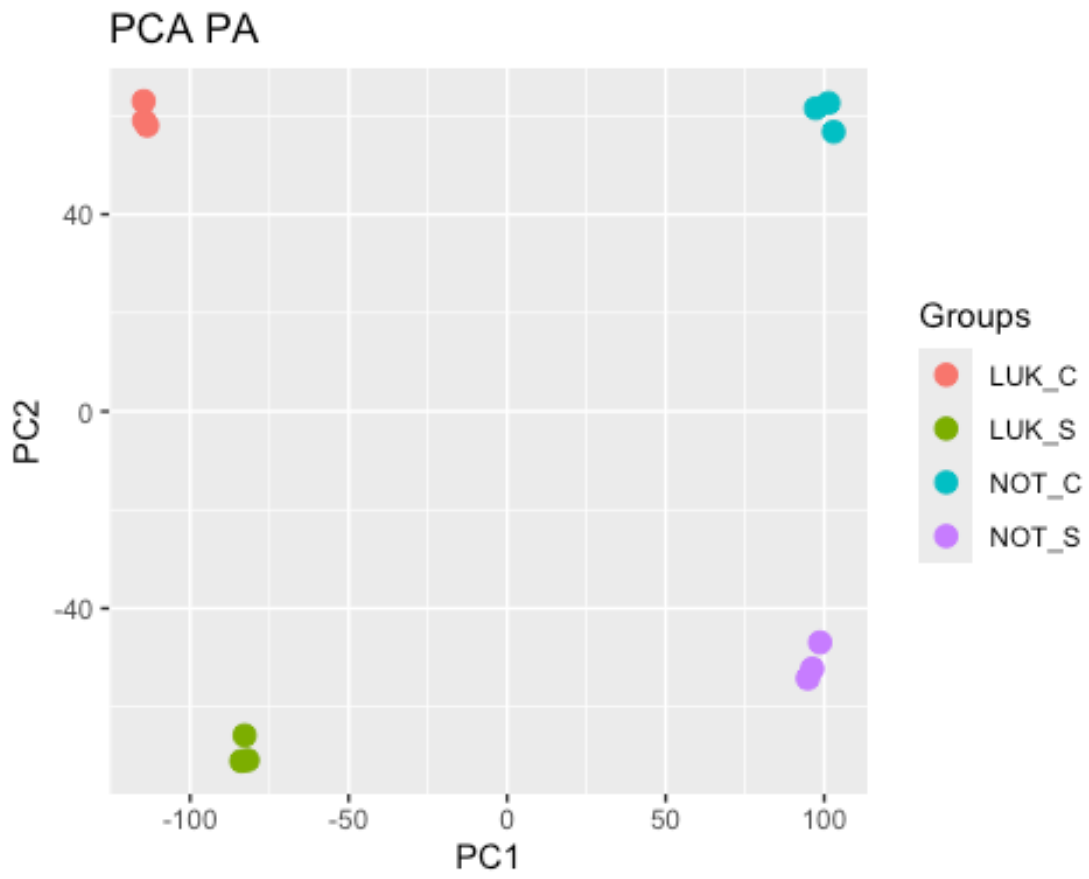
```r
PA_dataframe <- PA_dataframe[, -index_columns_NA]
```

Realitzem el PCA i visualitzem el resultat.

```r
# Filter the constant columns
PA_dataframe <- PA_dataframe[, apply(PA_dataframe, 2, var) != 0]
# Scale the data
PA_df_scaled <- scale(PA_dataframe)

# Perform the PCA
PCA_PA <- prcomp(PA_df_scaled, center = TRUE, scale. = TRUE)

# Summary of the result
summary(PCA_PA)
```

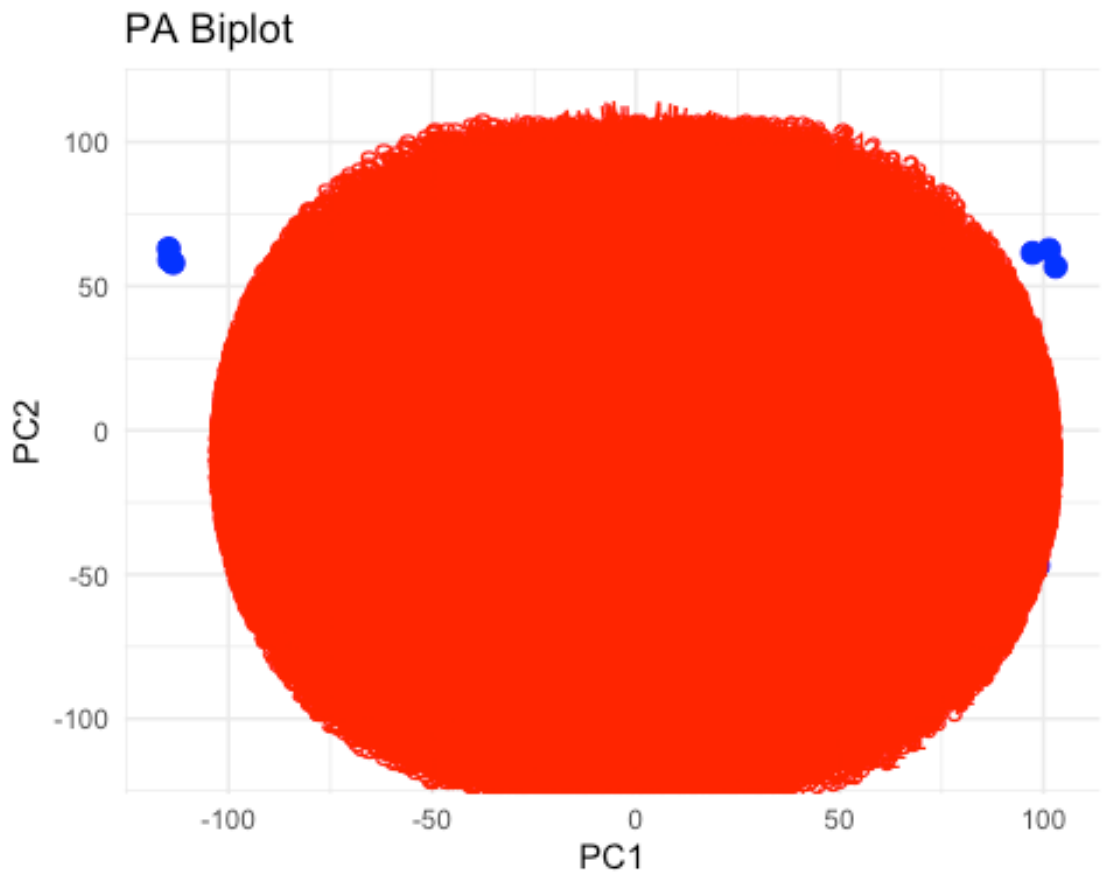
```
## Importance of components:
##                             PC1     PC2     PC3      PC4      PC5      PC6
## Standard deviation     103.6181 63.2389 54.6718 37.65985 36.42444 35.50776
## Proportion of Variance   0.3942  0.1468  0.1097  0.05207  0.04871  0.04629
## Cumulative Proportion    0.3942  0.5410  0.6507  0.70278  0.75149  0.79778
##                             PC7      PC8      PC9     PC10    PC11
PC12
## Standard deviation     34.94140 33.22045 32.94994 32.47970 32.2985 1.006e-
12
## Proportion of Variance  0.04482  0.04052  0.03986  0.03873  0.0383
0.000e+00
## Cumulative Proportion   0.84260  0.88312  0.92297  0.96170  1.0000
1.000e+00
```

```r
# Dataframe with the PCA result and the groups variable
PCA_PA_df <- data.frame(PC1 = PCA_PA$x[,1], PC2 = PCA_PA$x[,2], Group =
groups_PA)

# Graphic
ggplot(PCA_PA_df, aes(x = PC1, y = PC2, color = groups_PA)) +
  geom_point(size = 3) +
  labs(title = "PCA PA", x = "PC1", y = "PC2", color = "Groups")
```



Realització d'un biplot:

```
scores <- as.data.frame(PCA_PA$x)
loadings <- as.data.frame(PCA_PA$rotation)

# Scale the loadings so that they are displayed correctly in the biplot
scale_factor <- max(abs(scores$PC1), abs(scores$PC2)) /
max(abs(loadings$PC1), abs(loadings$PC2))
loadings <- loadings * scale_factor

# Create the biplot with ggplot2
ggplot() +
  geom_point(data = scores, aes(x = PC1, y = PC2), color = "blue", size = 3)
+  # Samples
  geom_segment(data = loadings, aes(x = 0, y = 0, xend = PC1, yend = PC2),
               arrow = arrow(length = unit(0., "cm")), color = "red") +  #
Variables arrows
   geom_text(data = loadings, aes(x = PC1, y = PC2, label =
rownames(loadings)),
          color = "red", vjust = 1.5) +  # variables Labels
  labs(title = "PA Biplot", x = "PC1", y = "PC2") +
  theme_minimal()
```



AR

Creem un dataframe que contingui les dades de AR tant de Lukullus com Notabillis i dels 3 tipus d'anàlisis. Les variables en columnes i les mostres en files.

Les dades fisiològiques estan mesurades en PA però reflecteixen l'estat global de tota la planta, per la qual cosa també les incorporarem en aquest anàlisi.

```
# Create a large datadrame with all the AR data
AR_dataframe <- cbind(physiological, transcriptomics_AR, metabolites_AR)
dim(AR_dataframe)

## [1]    12 29254
```

Modifiquem el dataframe de manera adequada per a realitzar el PCA.

```
# Save group names
groups_AR <- row.names(AR_dataframe)
print(groups_AR)

##  [1] "LUK CT1/CT2"  "LUK CT3/CT4"  "LUK CT4/CT5"  "LUK ST1/ST2"  "LUK
ST3/ST4"
##  [6] "LUK ST9/ST10" "NOT CT1/CT2"  "NOT CT3/CT4"  "NOT CT5/CT6"  "NOT
```

```
ST1/ST2"
## [11] "NOT ST3/ST4"  "NOT ST9"
```

```r
groups_AR <- gsub(".*(LUK|NOT).*(C|S).*", "\\1_\\2", groups_AR)
print(groups_AR)
```

```
##  [1] "LUK_C" "LUK_C" "LUK_C" "LUK_S" "LUK_S" "LUK_S" "NOT_C" "NOT_C"
"NOT_C"
## [10] "NOT_S" "NOT_S" "NOT_S"
```

```r
# Convert the dataframe to numeric values
AR_dataframe <- apply(AR_dataframe, 2, as.numeric)
row.names(AR_dataframe) <- groups_AR

# Delete columns with NA
columns_NA <- colSums(is.na(AR_dataframe)) > 0
index_columns_NA <- which(columns_NA)
print(index_columns_NA)
```

```
## LWP
##   3
```

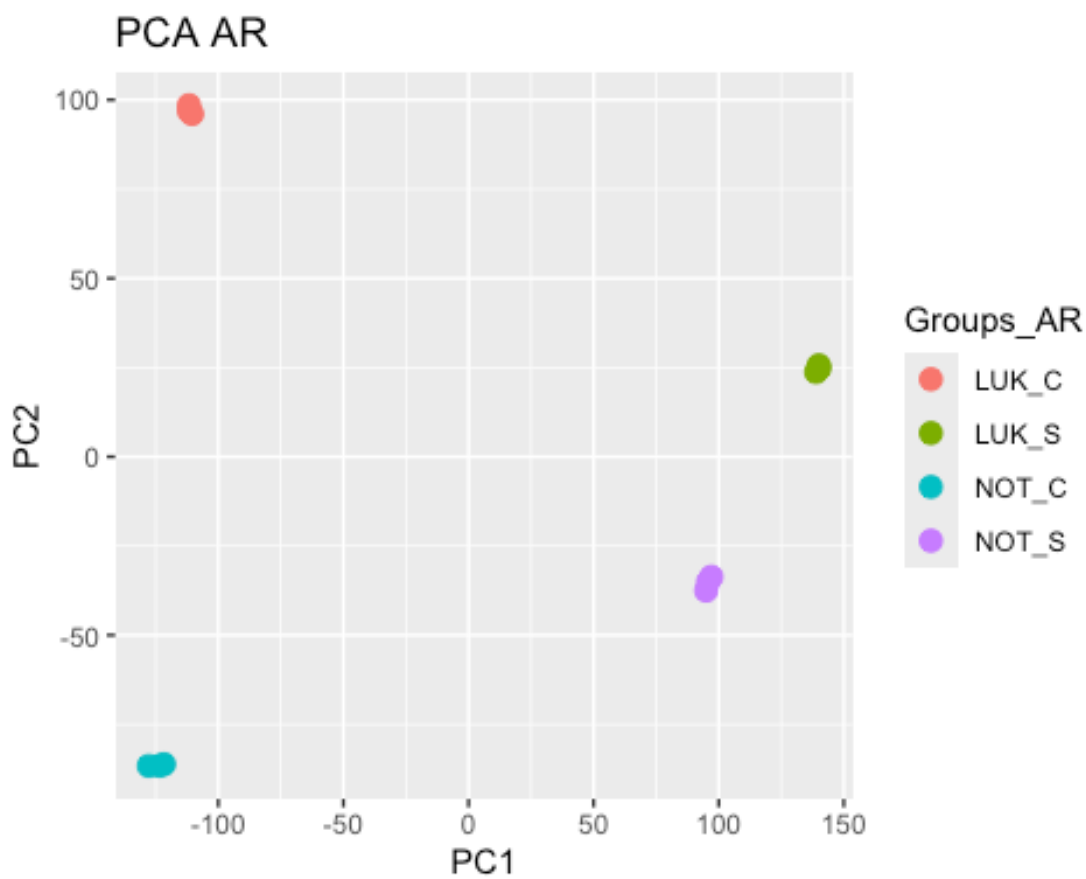
```r
AR_dataframe <- AR_dataframe[, -index_columns_NA]
```

Realitzem el PCA i visualitzem el resultat.

```r
# Filter the constant columns
AR_dataframe <- AR_dataframe[, apply(AR_dataframe, 2, var) != 0]
# Scale the data
AR_df_scaled <- scale(AR_dataframe)

# Perform the PCA
PCA_AR <- prcomp(AR_df_scaled, center = TRUE, scale. = TRUE)

# Summary of the result
summary(PCA_AR)
```

```
## Importance of components:
##                             PC1     PC2      PC3      PC4      PC5
PC6
## Standard deviation     124.2251 71.5732 51.25760 28.09033 26.89234
26.28603
## Proportion of Variance   0.5459  0.1812  0.09294  0.02791  0.02558
0.02444
## Cumulative Proportion    0.5459  0.7271  0.82002  0.84793  0.87351
0.89795
##                             PC7      PC8      PC9     PC10     PC11
PC12
## Standard deviation     26.08676 24.26880 23.86219 23.42626 22.29716 1.42e-
12
## Proportion of Variance  0.02407  0.02083  0.02014  0.01941  0.01759
```

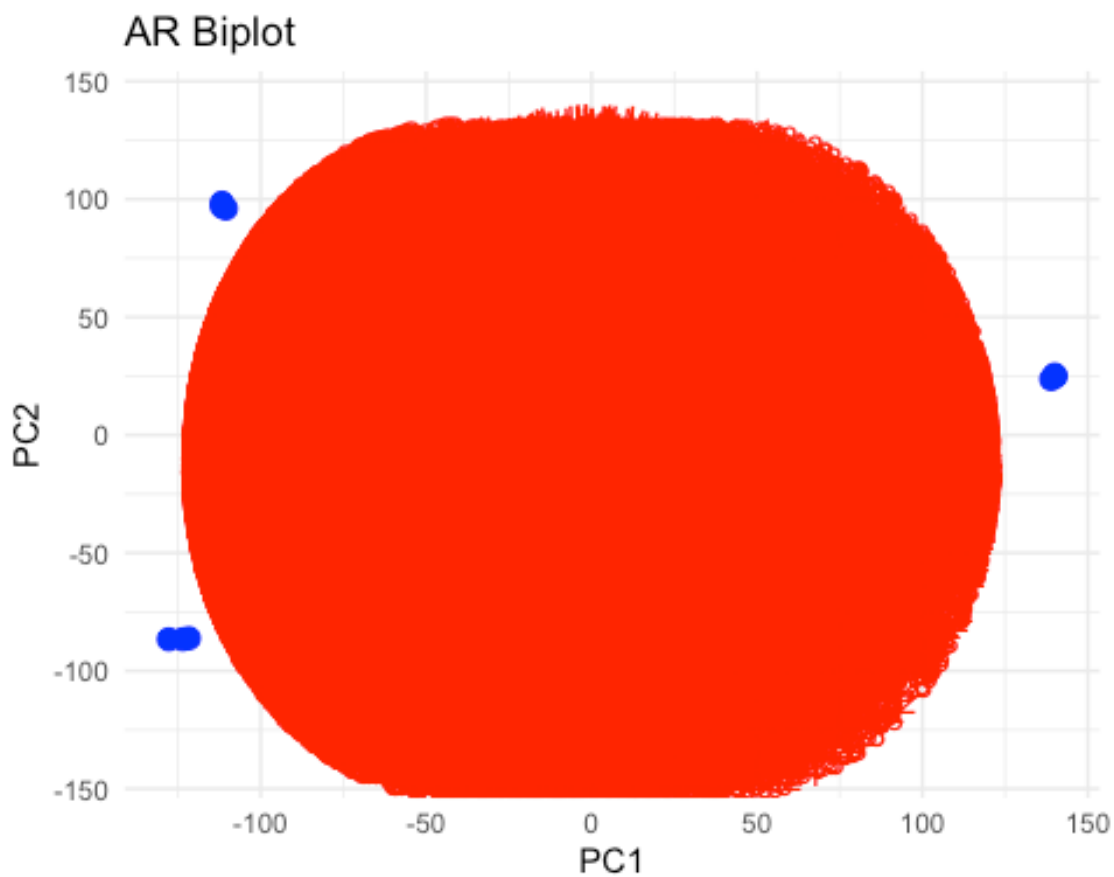
```
0.00e+00
## Cumulative Proportion   0.92203  0.94286  0.96300  0.98241  1.00000
1.00e+00

# Dataframe with the PCA result and the groups variable
PCA_AR_df <- data.frame(PC1 = PCA_AR$x[,1], PC2 = PCA_AR$x[,2], Group =
groups_AR)

# Crear el gráfico
ggplot(PCA_AR_df, aes(x = PC1, y = PC2, color = Group)) +
  geom_point(size = 3) +
  labs(title = "PCA AR", x = "PC1", y = "PC2", color = "Groups_AR")
```



Realització d'un biplot:

```
scores <- as.data.frame(PCA_AR$x)
loadings <- as.data.frame(PCA_AR$rotation)

# Scale the loadings so that they are displayed correctly in the biplot
scale_factor <- max(abs(scores$PC1), abs(scores$PC2)) /
max(abs(loadings$PC1), abs(loadings$PC2))
loadings <- loadings * scale_factor
```

```r
# Create the biplot with ggplot2
ggplot() +
  geom_point(data = scores, aes(x = PC1, y = PC2), color = "blue", size = 3)
+  # Samples
  geom_segment(data = loadings, aes(x = 0, y = 0, xend = PC1, yend = PC2),
               arrow = arrow(length = unit(0., "cm")), color = "red") +  #
Variables arrows
   geom_text(data = loadings, aes(x = PC1, y = PC2, label =
rownames(loadings)),
          color = "red", vjust = 1.5) +  # variables Labels
  labs(title = "AR Biplot", x = "PC1", y = "PC2") +
  theme_minimal()
```



PCA Segons genotip

Lukullus

Creem un dataframe que contingui les dades de Lukullus tant de PA com de AR i dels 3 tipus d'anàlisis. Les variables en columnes i les mostres en files.

```r
# Create a large datadrame with all the LUK data
Luk_dataframe <- cbind(physiological_LUK, transcriptomics_Luk,
metabolites_LUK)

## Warning in data.frame(..., check.names = FALSE): row names were found from
a
## short variable and have been discarded

rownames(Luk_dataframe) <- rownames(transcriptomics_Luk)
dim(Luk_dataframe)

## [1]    12 29169
```

Modifiquem el dataframe de manera adequada per a realitzar el PCA.

```r
# Save group names
groups_LUK <- row.names(Luk_dataframe)
print(groups_LUK)

##  [1] "LukullusAR.C7_1"  "LukullusAR.C8_1"  "LukullusAR.C9_1"
"LukullusAR.S10_1"
##  [5] "LukullusAR.S11_1" "LukullusAR.S12_1" "LukullusPA.C13_1"
"LukullusPA.C14_1"
##  [9] "LukullusPA.C15_1" "LukullusPA.S16_1" "LukullusPA.S17_1"
"LukullusPA.S18_1"

groups_LUK <- gsub(".*(AR|PA).*(C|S).*", "\\1_\\2", groups_LUK)
print(groups_LUK)

##  [1] "AR_C" "AR_C" "AR_C" "AR_S" "AR_S" "AR_S" "PA_C" "PA_C" "PA_C" "PA_S"
## [11] "PA_S" "PA_S"

# Convert the dataframe to numeric values
Luk_dataframe <- apply(Luk_dataframe, 2, as.numeric)
row.names (Luk_dataframe) <- groups_LUK

# Delete columns with NA
columns_NA <- colSums(is.na(Luk_dataframe)) > 0
index_columns_NA <- which(columns_NA)
print(index_columns_NA)

## LWP
##   3

Luk_dataframe <- Luk_dataframe[, -index_columns_NA]
```

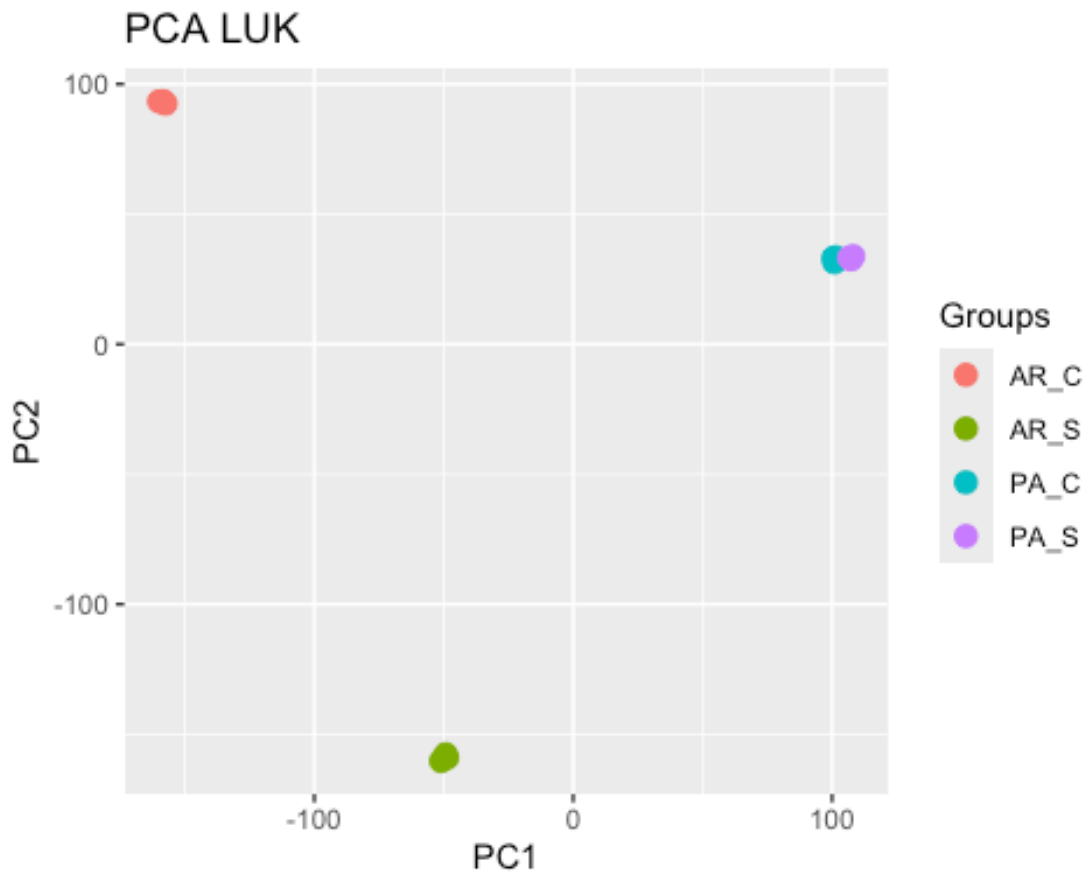
Realitzem el PCA i visualitzem el resultat.

```r
# Filter the constant columns
Luk_dataframe <- Luk_dataframe[, apply(Luk_dataframe, 2, var) != 0]
# Scale the data
Luk_df_scaled <- scale(Luk_dataframe)
```

```r
# Perform the PCA
PCA_LUK <- prcomp(Luk_df_scaled, center = TRUE, scale. = TRUE)

# Summary of the result
summary(PCA_LUK)
```

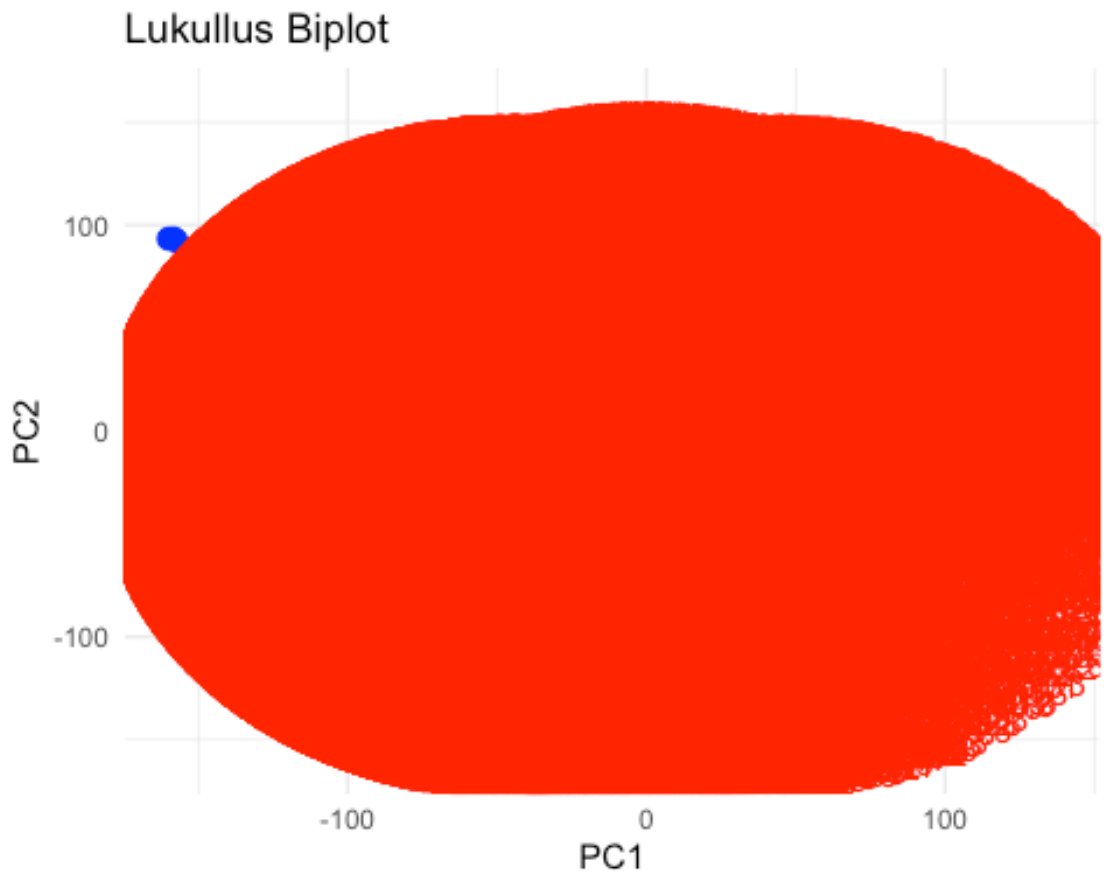
```
## Importance of components:
##                            PC1     PC2      PC3      PC4      PC5      PC6
## Standard deviation     115.978 99.2671 30.80245 25.71304 25.01591 22.62131
## Proportion of Variance   0.477  0.3495  0.03365  0.02345  0.02219  0.01815
## Cumulative Proportion    0.477  0.8265  0.86012  0.88357  0.90576  0.92391
##                             PC7      PC8      PC9     PC10     PC11
PC12
## Standard deviation     21.64813 21.37987 20.89197 20.04127 19.53844
1.227e-12
## Proportion of Variance  0.01662  0.01621  0.01548  0.01424  0.01354
0.000e+00
## Cumulative Proportion   0.94053  0.95674  0.97222  0.98646  1.00000
1.000e+00
```

```r
# Dataframe with the PCA result and the groups variable
PCA_LUK_df <- data.frame(PC1 = PCA_LUK$x[,1], PC2 = PCA_LUK$x[,2], Group =
groups_LUK)

# Crear el gráfico
ggplot(PCA_LUK_df, aes(x = PC1, y = PC2, color = groups_LUK)) +
  geom_point(size = 3) +
  labs(title = "PCA LUK", x = "PC1", y = "PC2", color = "Groups")
```



Realització d'un biplot:

```
scores <- as.data.frame(PCA_LUK$x)
loadings <- as.data.frame(PCA_LUK$rotation)

# Scale the loadings so that they are displayed correctly in the biplot
scale_factor <- max(abs(scores$PC1), abs(scores$PC2)) /
max(abs(loadings$PC1), abs(loadings$PC2))
loadings <- loadings * scale_factor

# Create the biplot with ggplot2
ggplot() +
  geom_point(data = scores, aes(x = PC1, y = PC2), color = "blue", size = 3)
+  # Samples
  geom_segment(data = loadings, aes(x = 0, y = 0, xend = PC1, yend = PC2),
               arrow = arrow(length = unit(0., "cm")), color = "red") +  #
Variables arrows
   geom_text(data = loadings, aes(x = PC1, y = PC2, label =
rownames(loadings)),
          color = "red", vjust = 1.5) +  # variables Labels
  labs(title = "Lukullus Biplot", x = "PC1", y = "PC2") +
  theme_minimal()
```



Notabilis

Creem un dataframe que contingui les dades de Notabilus tant de PA com de AR i dels 3 tipus d'anàlisis. Les variables en columnes i les mostres en files.

```
# Create a large datadrame with all the LUK data
Not_dataframe <- cbind(physiological_NOT, transcriptomics_Not,
metabolites_NOT)

## Warning in data.frame(..., check.names = FALSE): row names were found from
a
## short variable and have been discarded

rownames(Not_dataframe) <- rownames(transcriptomics_Not)
dim(Not_dataframe)

## [1]    12 29169
```

Modifiquem el dataframe de manera adequada per a realitzar el PCA.

```
# Save group names
groups_NOT <- row.names(Not_dataframe)
print(groups_NOT)
```

```
##  [1] "LukullusAR.C7_1"  "LukullusAR.C8_1"  "LukullusAR.C9_1"
"LukullusAR.S10_1"
##  [5] "LukullusAR.S11_1" "LukullusAR.S12_1" "LukullusPA.C13_1"
"LukullusPA.C14_1"
##  [9] "LukullusPA.C15_1" "LukullusPA.S16_1" "LukullusPA.S17_1"
"LukullusPA.S18_1"
```

```
groups_NOT <- gsub(".*(AR|PA).*(C|S).*", "\\1_\\2", groups_NOT)
print(groups_NOT)
```

```
##  [1] "AR_C" "AR_C" "AR_C" "AR_S" "AR_S" "AR_S" "PA_C" "PA_C" "PA_C" "PA_S"
## [11] "PA_S" "PA_S"
```

```
# Convert the dataframe to numeric values
Not_dataframe <- apply(Not_dataframe, 2, as.numeric)
row.names (Not_dataframe) <- groups_NOT

# Delete columns with NA
columns_NA <- colSums(is.na(Not_dataframe)) > 0
index_columns_NA <- which(columns_NA)
print(index_columns_NA)
```

```
## LWP
##   3
```

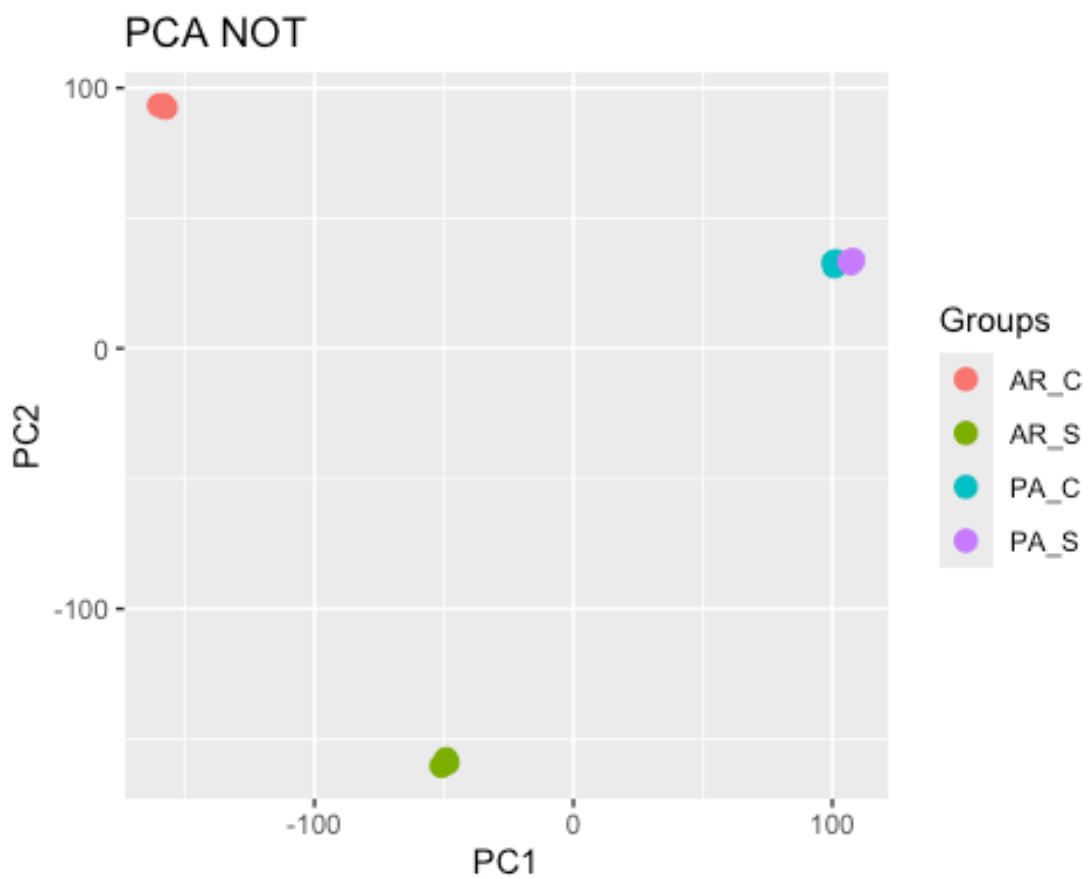
```
Not_dataframe <- Not_dataframe[, -index_columns_NA]
```

Realitzem el PCA i visualitzem el resultat.

```
# Filter the constant columns
Not_dataframe <- Not_dataframe[, apply(Not_dataframe, 2, var) != 0]
# Scale the data
Not_df_scaled <- scale(Not_dataframe)

# Perform the PCA
PCA_NOT <- prcomp(Not_df_scaled, center = TRUE, scale. = TRUE)

# Summary of the result
summary(PCA_NOT)
```

```
## Importance of components:
##                            PC1     PC2      PC3      PC4      PC5     PC6
## Standard deviation     115.9538 99.2583 30.84061 25.71236 25.06562 22.6539
## Proportion of Variance   0.4768  0.3494  0.03373  0.02345  0.02228  0.0182
## Cumulative Proportion    0.4768  0.8262  0.85994  0.88339  0.90567  0.9239
##                            PC7      PC8     PC9     PC10     PC11
PC12
## Standard deviation     21.66297 21.38569 20.9032 20.02633 19.54817 8.044e-
13
## Proportion of Variance  0.01664  0.01622  0.0155  0.01422  0.01355
0.000e+00
```

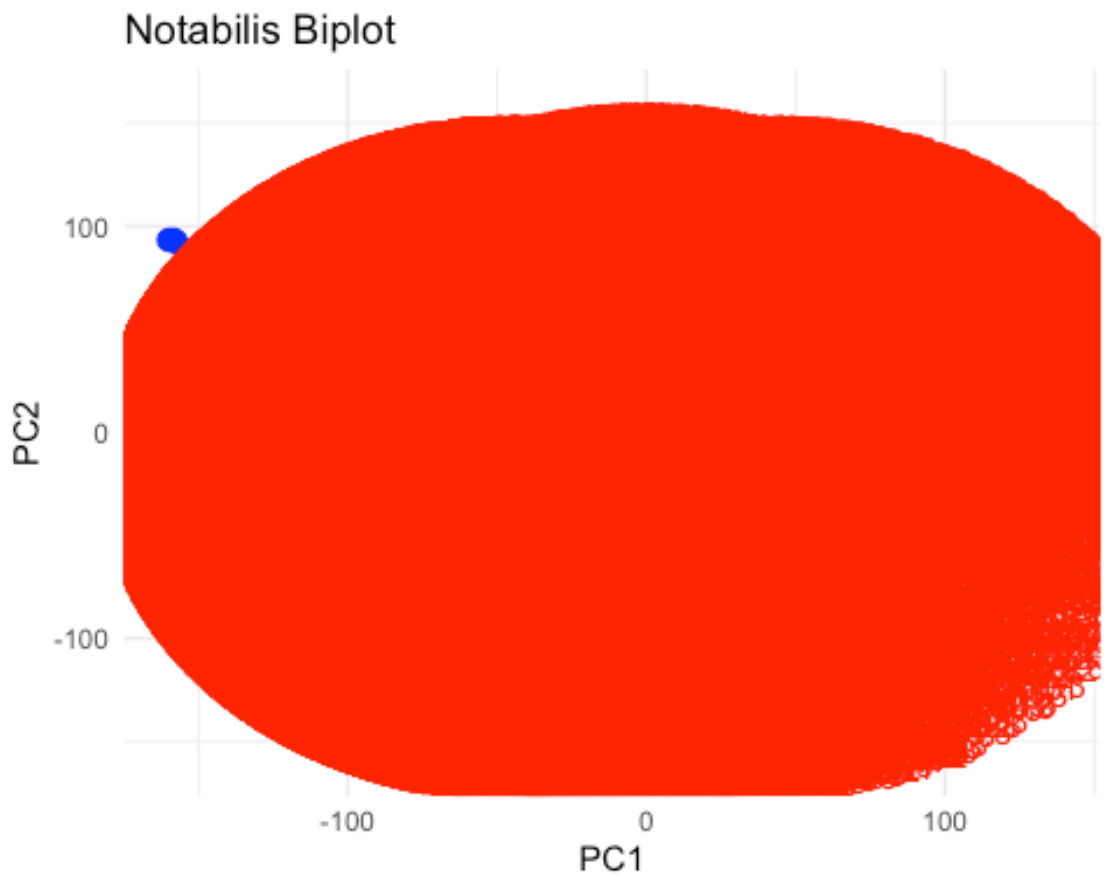
```
## Cumulative Proportion   0.94051  0.95673  0.9722  0.98645  1.00000
1.000e+00

# Dataframe with the PCA result and the groups variable
PCA_NOT_df <- data.frame(PC1 = PCA_NOT$x[,1], PC2 = PCA_NOT$x[,2], Group =
groups_NOT)

# Crear el gráfico
ggplot(PCA_NOT_df, aes(x = PC1, y = PC2, color = groups_NOT)) +
  geom_point(size = 3) +
  labs(title = "PCA NOT", x = "PC1", y = "PC2", color = "Groups")
```



Realització d'un biplot:

```
scores <- as.data.frame(PCA_LUK$x)
loadings <- as.data.frame(PCA_LUK$rotation)

# Scale the loadings so that they are displayed correctly in the biplot
scale_factor <- max(abs(scores$PC1), abs(scores$PC2)) /
max(abs(loadings$PC1), abs(loadings$PC2))
loadings <- loadings * scale_factor

# Create the biplot with ggplot2
```

```r
ggplot() +
  geom_point(data = scores, aes(x = PC1, y = PC2), color = "blue", size = 3)
+  # Samples
  geom_segment(data = loadings, aes(x = 0, y = 0, xend = PC1, yend = PC2),
               arrow = arrow(length = unit(0., "cm")), color = "red") +  #
Variables arrows
   geom_text(data = loadings, aes(x = PC1, y = PC2, label =
rownames(loadings)),
          color = "red", vjust = 1.5) +  # variables Labels
  labs(title = "Notabilis Biplot", x = "PC1", y = "PC2") +
  theme_minimal()
```



#

Exportació dels resultats

Guardem els gràfics en format png:

```r
library(gridExtra)

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
png(file="~/Desktop/PCA/PCA1_plots.png", width=600, height=350)

par(mfrow=c(2,2))

plot1<- ggplot(PCA_PA_df, aes(x = PC1, y = PC2, color = groups_PA)) +
  geom_point(size = 3) +
  labs(title = "PCA PA", x = "PC1", y = "PC2", color = "Groups")

plot2 <- ggplot(PCA_AR_df, aes(x = PC1, y = PC2, color = groups_AR)) +
  geom_point(size = 3) +
  labs(title = "PCA AR", x = "PC1", y = "PC2", color = "Groups")

plot3 <- ggplot(PCA_LUK_df, aes(x = PC1, y = PC2, color = groups_LUK)) +
  geom_point(size = 3) +
  labs(title = "PCA LUK", x = "PC1", y = "PC2", color = "Groups")

plot4 <- ggplot(PCA_NOT_df, aes(x = PC1, y = PC2, color = groups_NOT)) +
  geom_point(size = 3) +
  labs(title = "PCA NOT", x = "PC1", y = "PC2", color = "Groups")

grid.arrange(plot1, plot2, plot3, plot4, ncol = 2, nrow = 2)

dev.off()

## quartz_off_screen
##                 2
```