

Unidad 9: Visualización de resultados

Olga Andrea Hernández Miranda

2024-06-14

Existen diferentes tipos de gráficos para representar datos de RNAseq:
-Expresión diferencial: -Volcano plot
-Diagrama de Venn
-Gráficas de barras
-Perfil de expresión: Heatmap
-Enriquecimiento funcional: Análisis de enriquecimiento funcional
Bubble plot

Directorio de trabajo

```
directorio <- "C:/Users/andii/OneDrive/Documents/02Fun-R-transcript/data"  
setwd(directorio)
```

Volcano plot Los datos de expresión diferencial se pueden visualizar a través de una gráfica de volcán en la cual el eje x representa el fold change, mientras que el eje y representa al p-value.

```
#BiocManager::install('EnhancedVolcano')  
#install.packages('BiocManager')  
library(BiocManager)
```



```
## Bioconductor version '3.16' is out-of-date; the current release version '3.19'  
##   is available with R version '4.4'; see https://bioconductor.org/install
```



```
library(EnhancedVolcano)
```

```
## Loading required package: ggplot2  
  
## Loading required package: ggrepel
```



```
library("RColorBrewer")
```

Abrimos el archivo con datos de expresión diferencial para graficarlos

```
resED <- read.table("U9_tabla_ed_crudos_v2.csv", sep = ",", header = T)  
head(resED)
```

```
##          X sampleA sampleB  baseMeanA baseMeanB  baseMean  
## 1 ENSG00000001084.6  ETAPA1  ETAPA2    1.991894  4350.945  2176.4686  
## 2 ENSG00000005175.5  ETAPA1  ETAPA2    1.224946  1825.860   913.5422  
## 3 ENSG00000001036.8  ETAPA1  ETAPA2    4.670784  1000.665   502.6677  
## 4 ENSG00000000419.8  ETAPA1  ETAPA2    9.719613  1056.972   533.3456  
## 5 ENSG00000005189.15 ETAPA1  ETAPA2    0.000000  1364.821   682.4104  
## 6 ENSG00000002726.15 ETAPA1  ETAPA2  1046.388610 43147.426  22096.9073
```

```

##   log2FoldChange      lfcSE      stat    pvalue     padj
## 1    -11.112967 1.2075556 -9.202861 3.49e-20 1.27e-15
## 2    -10.505649 1.3440607 -7.816350 5.44e-15 9.02e-11
## 3    -7.730563 0.9940725 -7.776659 7.45e-15 9.02e-11
## 4    -6.739602 0.8774695 -7.680725 1.58e-14 1.44e-10
## 5   -13.367313 1.8580524 -7.194261 6.28e-13 4.56e-09
## 6    -5.366136 0.7618378 -7.043673 1.87e-12 1.13e-08

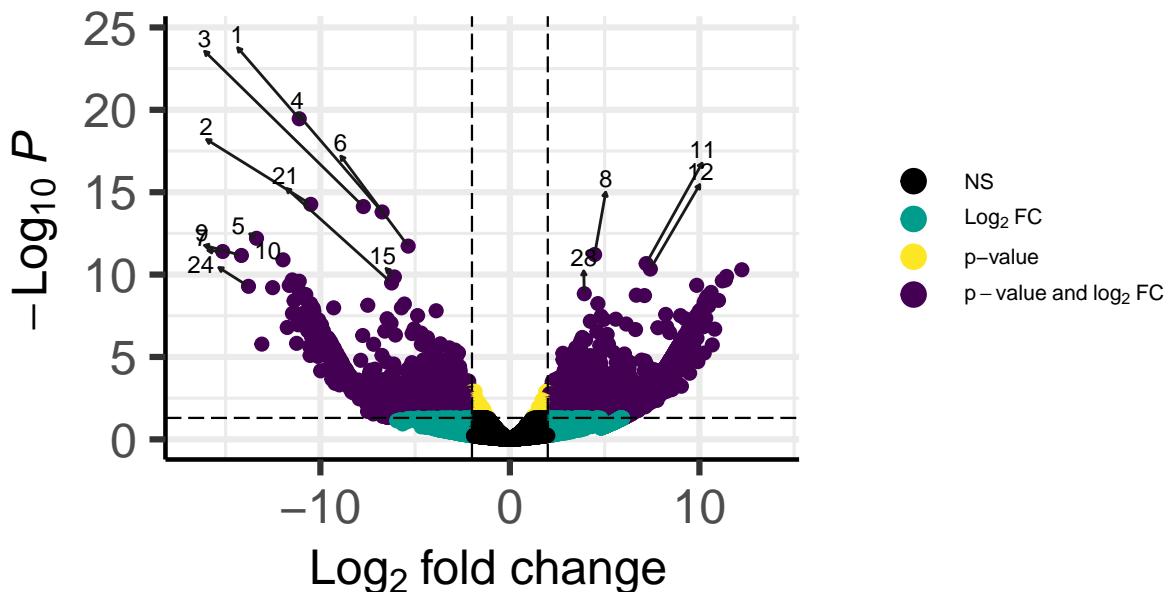
EnhancedVolcano(resED,
  lab = rownames(resED),
  x = 'log2FoldChange',
  y = 'pvalue',
  title = 'SAM vs MI',
  xlab = bquote(~Log[2]~ 'fold change'),
  pCutoff = 0.05,
  FCcutoff = 2.0,
  pointSize = 2.0,
  labSize = 3.0,
  col = c('black', '#009c8c', '#FDE725', '#440154'),
  colAlpha = 1,
  legendPosition = 'right',
  legendLabSize = 8,
  legendIconSize = 4.0,
  drawConnectors = T,
  widthConnectors = 0.5)

## Warning: ggrepel: 5942 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps

```

SAM vs MI

EnhancedVolcano



total = 41113 variables

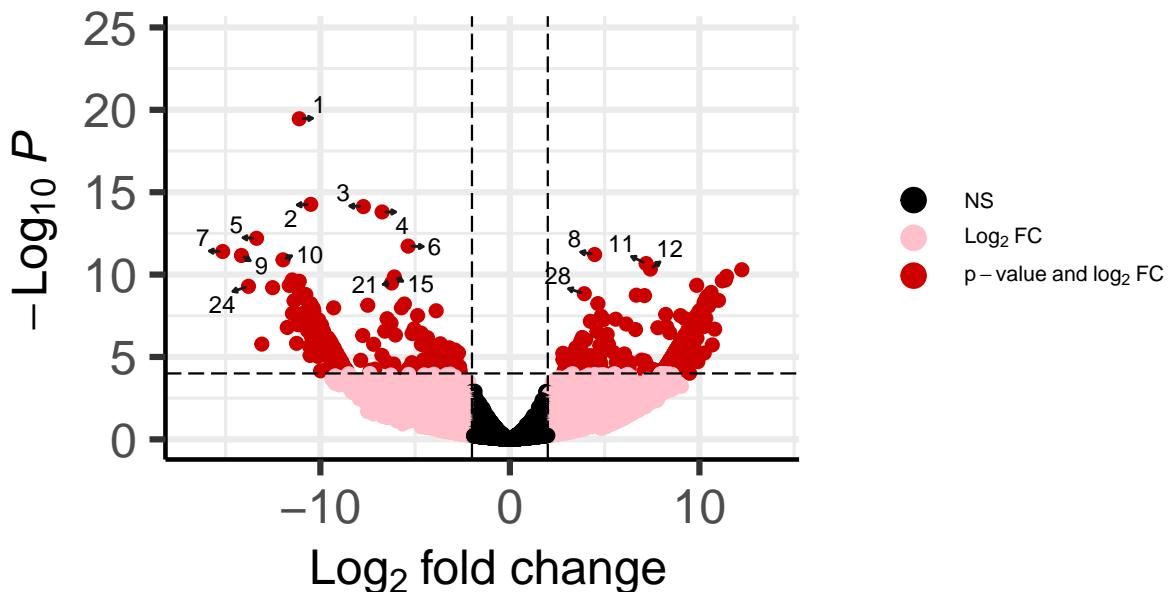
V2

```
EnhancedVolcano(resED,
  lab = rownames(resED),
  x = 'log2FoldChange',
  y = 'pvalue',
  title = 'SAM vs MI',
  xlab = bquote(~Log[2]~ 'fold change'),
  pCutoff = 10e-5,
  FCcutoff = 2.0,
  pointSize = 2.0,
  labSize = 3.0,
  col = c('black', 'pink', 'purple', 'red3'),
  colAlpha = 1,
  legendPosition = 'right',
  legendLabSize = 8,
  legendIconSize = 4.0,
  drawConnectors = T,
  widthConnectors = 0.5)
```

```
## Warning: ggrepel: 712 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```

SAM vs MI

EnhancedVolcano



total = 41113 variables

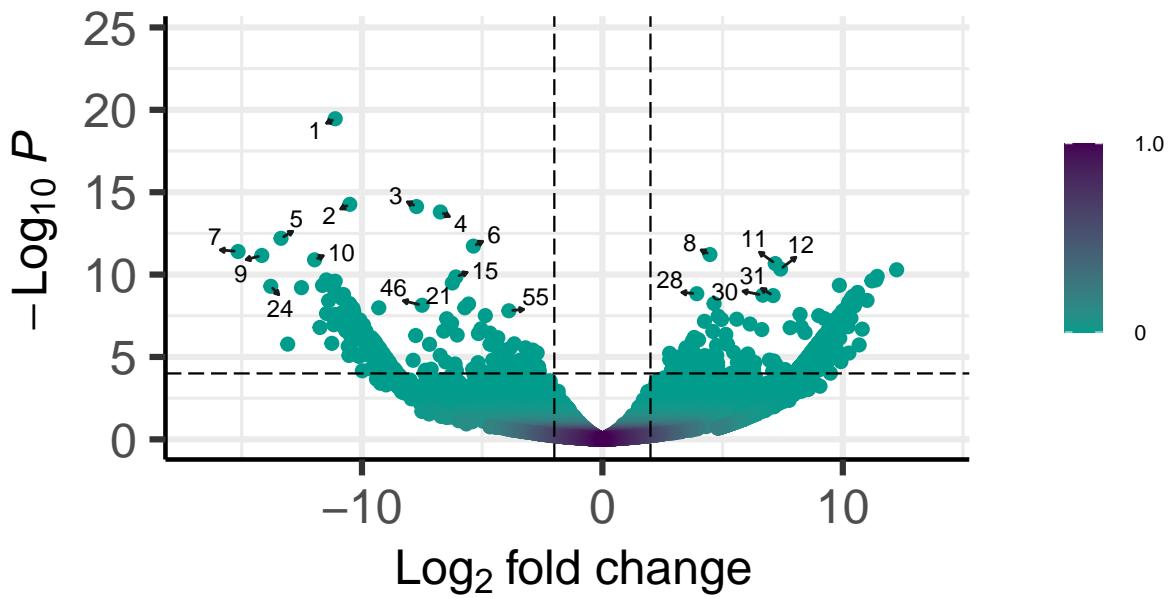
V3

```
EnhancedVolcano(resED,
  lab = rownames(resED),
  x = 'log2FoldChange',
  y = 'pvalue',
  title = 'SAM vs MI',
  xlab = bquote(~Log[2]~ 'fold change'),
  pCutoff = 10e-5,
  FCcutoff = 2.0,
  pointSize = 2.0,
  labSize = 3.0,
  colAlpha = 1,
  colGradient = c('#009c8c', '#440154'),
  legendPosition = 'right',
  legendLabSize = 8,
  legendIconSize = 4.0,
  drawConnectors = T,
  widthConnectors = 0.5)
```

```
## Warning: ggrepel: 708 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```

SAM vs MI

EnhancedVolcano



total = 41113 variables

Diagrama de Venn

Listas de genes con expresión diferencial para ver las coincidencias y diferencias.

Librerías

```
#install.packages("VennDiagram")
```

```
library(VennDiagram)
```

```
## Loading required package: grid
```

```
## Loading required package: futile.logger
```

Genes inducidos

```
SAM_vs_MI <- read.csv("U9_SAM_vs_MI_up.csv", sep = ",", header = TRUE)
head(SAM_vs_MI)
```

```
##      gene_id
## 1 AT1G77080
## 2 AT1G54830
## 3 AT1G61040
## 4 AT1G65480
## 5 AT1G25560
## 6 AT1G79730
```

```

MI_vs_MF <- read.csv("U9_MI_vs_MF_up.csv", sep = ",", header = TRUE)
head(MI_vs_MF)

##      gene_id
## 1 AT1G77080
## 2 AT1G54830
## 3 AT1G61040
## 4 AT1G65480
## 5 AT1G25560
## 6 AT1G79730

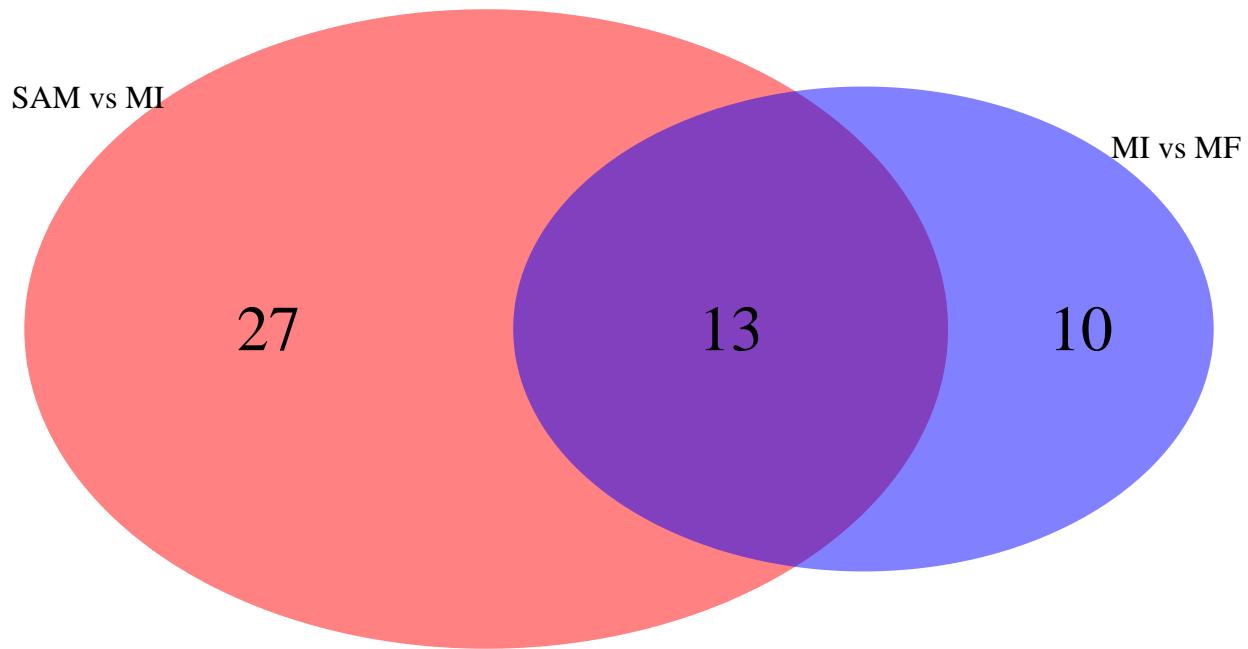
# Obtener los conjuntos de genes
genes_SAM_vs_MI <- SAM_vs_MI$gene_id
genes_MI_vs_MF <- MI_vs_MF$gene_id

# Graficar diagrama de Venn

venn.plot <- venn.diagram(
  x = list(SAM_vs_MI = genes_SAM_vs_MI, MI_vs_MF = genes_MI_vs_MF),
  category.names = c("SAM vs MI", "MI vs MF"),
  filename = NULL,
  output = T,
  col = "transparent", # Color transparente
  fill = c("red", "blue"), # Colores de los círculos
  alpha = 0.5, # Transparencia
  lwd = 2, # Ancho de los contornos
  cex = 2 # Tamaño del texto
)

# Mostrar diagrama de Venn
grid.draw(venn.plot)

```



Genes reprimidos

```
# Leer los archivos CSV
SAM_vs_MI <- read.csv("U9_SAM_vs_MI_down.csv", sep = ",", header = TRUE)
head(SAM_vs_MI)
```

```
##          gene_id
## 1    AT3G11435
## 2    AT2G34720
## 3    AT2G39250
## 4    AT3G22380
## 5    AT3G26640
## 6    AT3G18990
```

```
MI_vs_MF <- read.csv("U9_MI_vs_MF_down.csv", sep = ",", header = TRUE)
head(MI_vs_MF)
```

```
##          gene_id
## 1    AT3G11435
## 2    AT2G34720
## 3    AT2G39250
## 4    AT3G22380
## 5    AT3G26640
## 6    AT3G18990
```

```

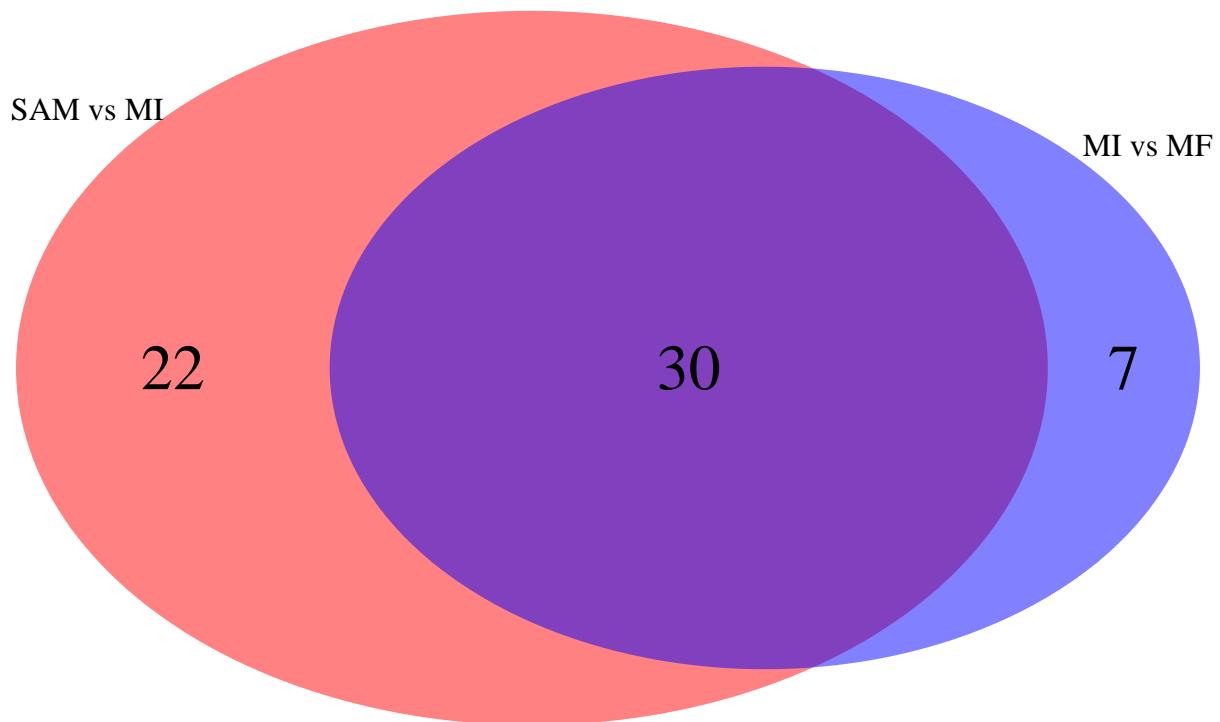
# Obtener los conjuntos de genes
genes_SAM_vs_MI <- SAM_vs_MI$gene_id
genes_MI_vs_MF <- MI_vs_MF$gene_id

# Graficar diagrama de Venn

venn.plot <- venn.diagram(
  x = list(SAM_vs_MI = genes_SAM_vs_MI, MI_vs_MF = genes_MI_vs_MF),
  category.names = c("SAM vs MI", "MI vs MF"),
  filename = NULL,
  output = TRUE,
  col = "transparent", # Establecer el color transparente
  fill = c("red", "blue"), # Colores de los círculos
  alpha = 0.5, # Transparencia
  lwd = 2, # Ancho de los contornos
  cex = 2 # Tamaño del texto
)

# Mostrar diagrama de Venn
grid.draw(venn.plot)

```



Gráfica de barras

Datos de genes inducidos o reprimidos

```

library(ggplot2)

Datos <- read.table("U9_Genes_up_down.csv", sep = ",", header = T)

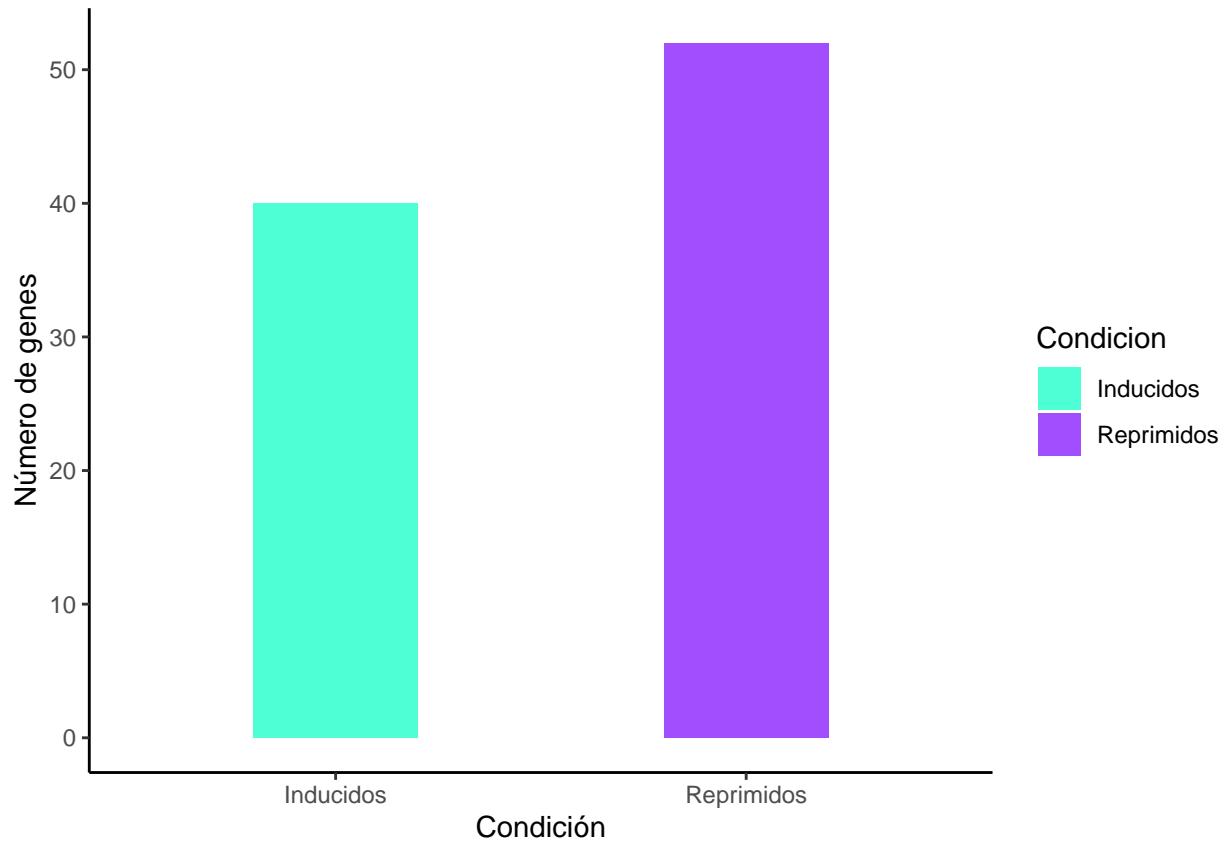
Datos

##      Condicion No_de_genes
## 1    Inducidos        40
## 2 Reprimidos        52

colores2 <- c("#00FFC3", "#7A00FF")

ggplot(Datos, aes(x = Condicion, y = No_de_genes, fill = Condicion)) +
  geom_bar(stat = 'identity', alpha = 0.7, width = 0.4) +
  theme_classic() +
  scale_fill_manual(values = colores2) +
  labs(x = "Condición", y = "Número de genes")

```



Heatmap

En el heatmap se representan datos de expresión en TPM. Se genera un análisis de conglomerados para identificar perfiles de expresión.

Librerías

```

library("gplots")

##
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
##
##     lowess

library("RColorBrewer")
library("viridis")

## Loading required package: viridisLite

library(lattice)
library(dendextend)

##
## -----
## Welcome to dendextend version 1.17.1
## Type citation('dendextend') for how to cite the package.
##
## Type browseVignettes(package = 'dendextend') for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues
## You may ask questions at stackoverflow, use the r and dendextend tags:
##     https://stackoverflow.com/questions/tagged/dendextend
##
## To suppress this message use: suppressPackageStartupMessages(library(dendextend))
## -----


##
## Attaching package: 'dendextend'

## The following object is masked from 'package:VennDiagram':
##
##     rotate

## The following object is masked from 'package:stats':
##
##     cutree

```

Crear matriz y normalizar datos

```

alpha <- 0.01
directorio <- "C:/Users/andii/OneDrive/Documents/02Fun-R-transcript/data"
setwd(directorio)
data <- read.table("U9_HM.csv", sep = ",", header = T)
row.names(data) <- data[,1]
mat_data <- data.matrix(data[,1:ncol(data)])
mat_data2 <- mat_data[,-1]
mat_data2

```

```

##      MAF1 NFYC9 VIP5  FT TEM1 ELF7 GA2ox7 PHYA SPA4 SEP3.. CAL miR159a UBC1
## SAM_R1  785   845   600    4   600   320    610   890   400    555   500    300   780
## SAM_R2  525   720   599    4   700   230    456   900   569    650   456    299   800
## MI_R1   2     6     3   560    7     5     10    20    9     14    13     8    30
## MI_R2   2     1     5   600    7     3     6    10    9     13    12     8    18
##      GA3ox2 FLD miR159c VIP6 LKP2 CLF miR172a ATGRP7 FES SPA3 SPL3 ELF4
## SAM_R1  210   520   250   199   189   148    160   186   100    48    130    98
## SAM_R2  205   250   280   200   198   201    150   177   100    7     129   100
## MI_R1   8     2     18    9     9     8     9     10    6     2     11    10
## MI_R2   7     26    2    10    10    12    10    15    8     3     14    15
##      miR156g MYB65 Cstf64 NFYC3.HAP5C GA3ox1 HUB1 CDF3 GA2ox3 GID1A CCA1 ESD1
## SAM_R1  8200   48   1797          8200   7800   348   489   2760   464   2712  2556
## SAM_R2  6100   44   1606          8000   8500   380   343   3204   473   2801  2521
## MI_R1   1333   14   736            3800   4005   219   279   2143   341   2459  2402
## MI_R2   1000   10   763            4000   4200   189   306   2125   334   2397  2219
##      FLK WNK1 PIE1 VIL1 miR172c NFYA4  SNZ TIC LWD2 VRN1 FPA SOC1 TEM2
## SAM_R1  5303  5675  7529  9179   6064  37252  17892  2122   892  1139  2627  1627  21268
## SAM_R2  5231  5649  7942  9066   5823  36389  18247  2086   921  1051  2422  1515  20987
## MI_R1   4821  5252  7348  8909   7000  43760  21910  2605  1160  1382  3392  2136  28942
## MI_R2   4909  5325  7337  8749   6842  43284  21359  2510  1180  1449  3228  2076  28239
##      SPA1 LUX.PCL1 SPL9 SPL5   GI LWD1.ATAN11 CRY2 COP1 GA2ox1 RFI2 CDF5 PHYB
## SAM_R1  1826        400  1132   170   500          489  1800   560    109   400   102   60
## SAM_R2  1620        463  1084   177   515          489  2000   389    100   148   99    60
## MI_R1   2139        644  1693   294   1800          2000  8000   2150   499  1586   555   350
## MI_R2   2564        580  1660   287   1990          2100  8000   1860   500  1452   600   360
##      LHY PIF3 GA2.KS NFYB1.HAP3A SAP18 TOE1 ATX1 GA2ox6 RGA miR156a FI01
## SAM_R1  999   160    29          235   400   250    12    450   500     9    8
## SAM_R2  1000  214    30          125   450   320    15    412   425    14    14
## MI_R1   7600  1458   300          1562  2300  4560   189   5600  5200   205   199
## MI_R2   7800  1585   299          2562  7500  2563   150   6300  8560   170   200
##      ELF3 PFT1 SDG26 UBC2 PRR9 SVP Cstf77 GA2ox2 GA2ox4 GAI SPL4 miR159b
## SAM_R1   6    13     8   20    10    7    16     6    12    15    9    11
## SAM_R2  13    20     8   8     15    2    15     10    4    10    4    9
## MI_R1   170   600   299   489   100   280    700    562   526   999   500   956
## MI_R2   189   499   300   600   999   150    986    562   610   1000  555   780
##      NFYC2.HAP5B AP1 PNF FKF1 AGL17 FVE HUB2 EFS
## SAM_R1      5    5     1    5     2    3     2    2
## SAM_R2      7    5     3    5     2    1     6    2
## MI_R1   452   620   253   456    300   299   652   785
## MI_R2   650   400   220   756    250   320   800   525

```

```

countTable.kept <- log2(mat_data2) #Se obtiene el log2 para reducir diferencias
dim(countTable.kept)

```

```

## [1] 4 92

datos <- countTable.kept

datos <- scale(datos)

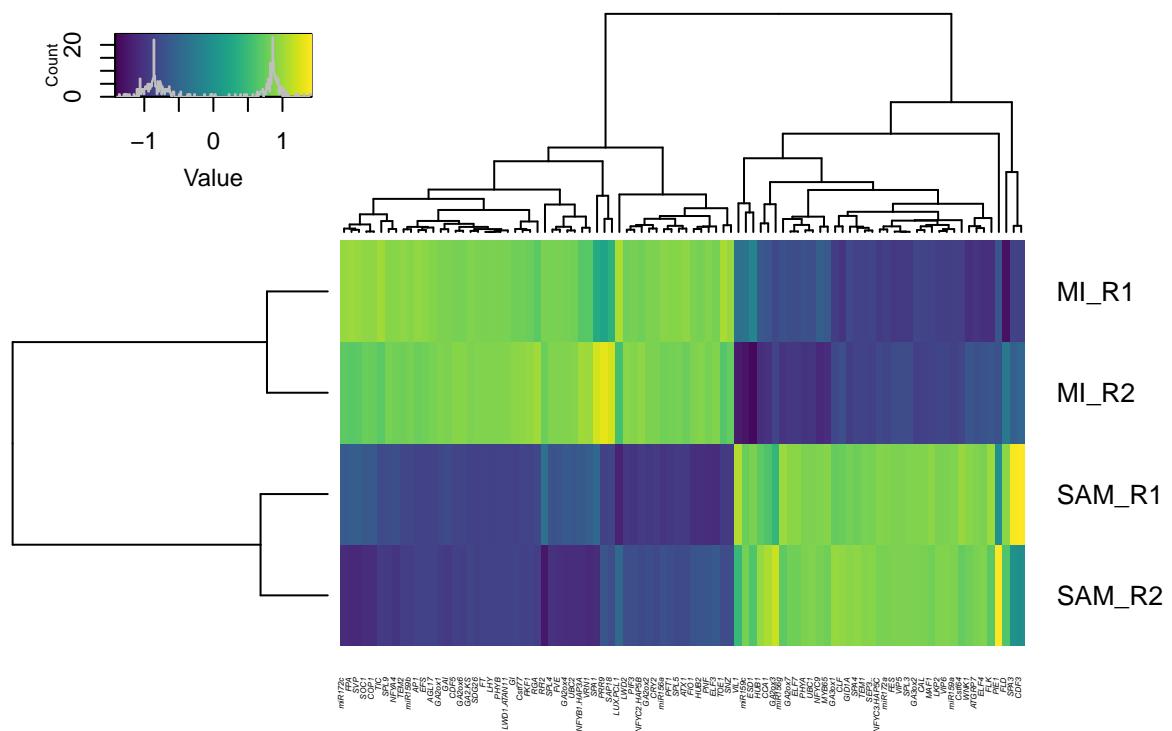
colores <- viridis(256)

colores2 <- magma(256)

```

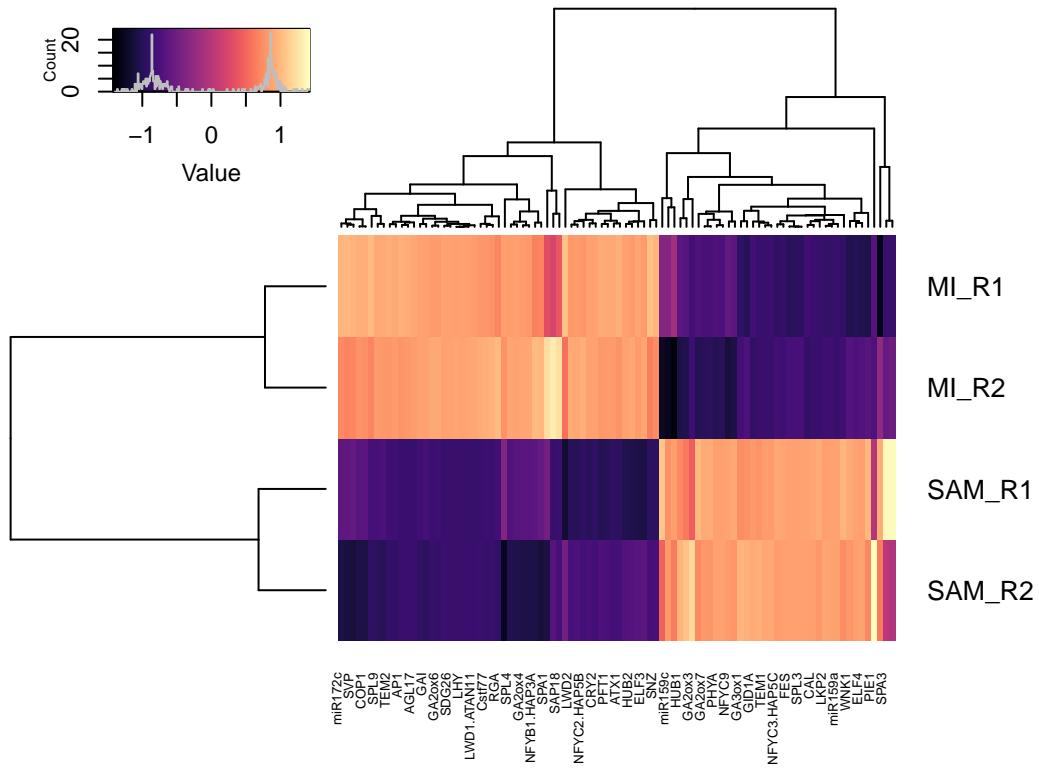
V1

```
heatmap.2(as.matrix (datos),
          cexCol = 0.3,
          cexRow = 1,
          labCol=as.expression(lapply(colnames(datos),function(a) bquote(italic(.(a))))),
          trace="none", hline = NA,
          margins =c(7,7),
          denscol = "grey",
          key.title=NA,
          col = colores
      )
```



V2

```
heatmap.2(as.matrix (datos),
          cexCol = 0.5,
          cexRow= 1,
          trace="none", hline = NA,
          margins =c(7,11),
          denscol = "grey",
          key.title=NA,
          col = colores2
      )
```



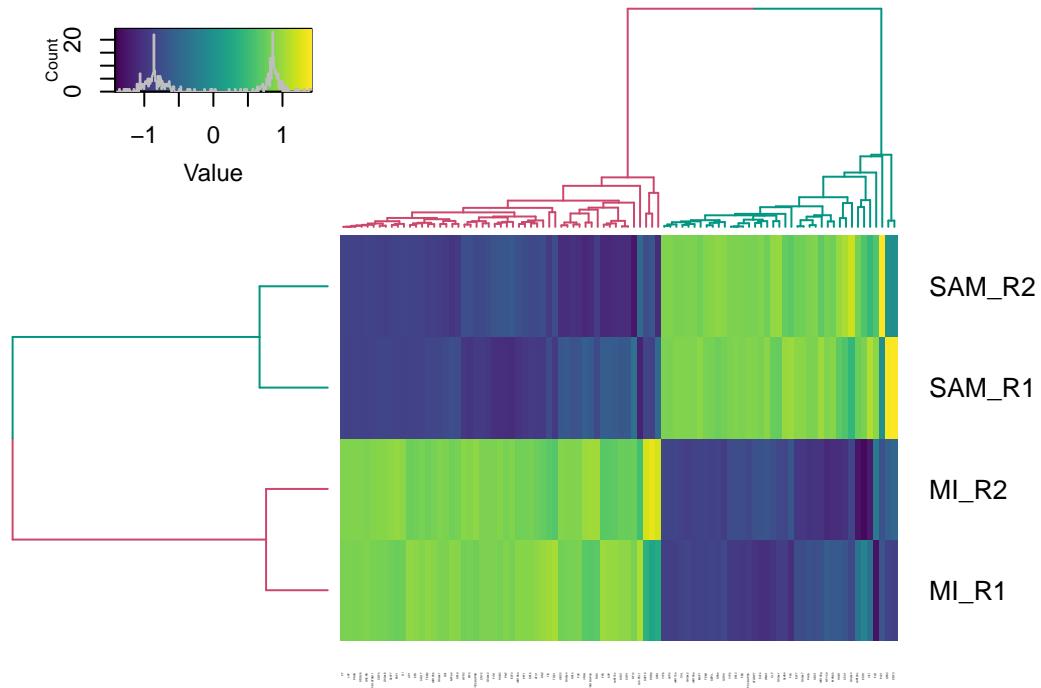
Versión con clusters

```
dend_r <- datos %>% dist(method = "euclidean") %>%
  hclust(method = "average") %>%
  as.dendrogram %>% ladderize %>%
  color_branches(k=2)

dend_c <- t(datos) %>% dist(method = "euclidean") %>%
  hclust(method = "average") %>%
  as.dendrogram %>% ladderize %>%
  color_branches(k=2)

#Version 2 con clusters

heatmap.2(as.matrix (datos),
  cexCol = 0.1,
  cexRow= 1,
  Rowv = dend_r,
  Colv = dend_c,
  trace="none", hline = NA,
  margins =c(7,11),
  denscol = "grey",
  key.title=NA,
  col = colores
)
```



Enriquecimiento funcional

Identificar grupos de genes o rutas metabólicas significativamente enriquecidas, se compara estadísticamente entre el experimento y una base de datos.

En este ejercicio solo se van a graficar los datos obtenidos de las bases de datos.

Librerías

```
library(ggplot2)
library(forcats)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr     1.1.4     v stringr    1.5.1
## v lubridate 1.9.3     v tibble     3.2.1
## v purrr     1.0.2     v tidyr     1.3.1
## v readr     2.1.5
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(dplyr)
library(viridis)
library(ggthemes)
```

Bubble plot

```
directorio <- "C:/Users/andii/OneDrive/Documents/02Fun-R-transcript/data"
setwd(directorio)

mydat <- read.table("U9_BP_bubble.csv", sep = ",", header = T)

head(mydat)

##                               GO_term      Value LogSize
## 1 alternative mRNA splicing, via spliceosome -18.051587 1.204120
## 2             response to acid chemical   -1.440093 2.625312
## 3           cell fate specification    -7.649752 1.518514
## 4           immune system process     -1.306273 2.408240
## 5 regulation of immune system process  -2.712198 2.049218
## 6 regulation of response to biotic stimulus -5.112383 2.324282

mydat <- mydat[order(mydat$Value, decreasing = TRUE), ]

top10 <- head(mydat, 10)

head(top10)

##                               GO_term      Value LogSize
## 272 cellular biosynthetic process -1.301899 3.609061
## 4       immune system process     -1.306273 2.408240
## 275 alcohol metabolic process  -1.320572 2.252853
## 273       response to salt     -1.324222 1.690196
## 255 intracellular chemical homeostasis -1.355561 2.318063
## 196 terpenoid biosynthetic process -1.366532 2.143015

# Reordenar el eje y basado en Log10Pvalue
top10$GO_term <- reorder(top10$GO_term, -top10$Value)

# Poner el orden al revés
top10$GO_term <- factor(top10$GO_term, levels = rev(levels(top10$GO_term)))

# Grafica de burbuja con paleta de colores viridis
ggplot(top10, aes(x = LogSize, y = GO_term)) +
  geom_point(aes(color = Value, size = LogSize)) +
  scale_color_viridis(option = "viridis") +
  theme_bw() +
  theme(axis.text.y = element_text(size = 10))
```

