ADO - PEC1 - Informe del análisis

José Félix Rojas Cabeza

2020 - April - 27

Contents

Abstract:

Se investigó la expresión diferencial de genes en células CML CD34+ sometidas a dos tratamientos en un modelo 2x2, Se realizó un workflow típico donde se obtuvieron, las listas de los genes diferencialmente expresados por cada tratamiento, volcano plots, un heatmap ordenado de los genes diferencialmente expresados, y mapas de redes. Los datos y el código para el análisis se proporcionan en un repositorio de github.

Objetivos:

Se pretende asimilar, a través de la práctica con un caso concreto, el proceso de análisis de microarrays siguiendo un workflow de ejemplo para analizar datos de microarrays utilizando paquetes de R y Bioconductor.

Se pretende ubicar los genes diferencialmente expresados que cumplan con:

$$FDR < 0.1 \wedge log_2|FC| \geq 1$$

est La intención de utilizar FDR es controlar el error tipo I. El Fold Change se utiliza para determinar cuáles son los p valores más bajos.

Materiales y métodos:

Datos:

El conexto del estudio (Zhang et al. 2013) es el siguiente: Se estaba investigando una forma de tratar la Leucemia Mieloide Crónica (CML), y la dificultad principal del uso de inhibidores de Tirosina-Linasa (TKI), presentaba dificultades debido a la incapacidad de eliminar Células Madre de Leucemia (LSC). El cocultivo con células estromales mesenquimales (MSC) de médula ósea humana inhibió significativamente la apoptosis y conservó las células madre / progenitoras de CML después de la exposición a TKI, manteniendo la capacidad de formación de colonias y el potencial de injerto en ratones inmunodeficientes. El paper presenta que la protección a las LSC opera con la intervención de N-cadherina y Wnt-β-catenina.

Microarrays:

Se extrajo ARN de células CML CD34+ tratadas con o sin imatinib (IM), un inhibidor de Kinasas, y Células Mesenquimales del Estroma (MSC), por 96 horas, amplifcadas, etiquetadas e hibridizadas, a arreglos GeneChip 1.0 (Affymetrix, Santa Clara, CA). Se compararon las interacciones de controles, tratamiento con IM, tratamiento con MSC, y ambos tratamientos, con 3 réplicas cada una. El arreglo (de un color) tenía 33297 sondas.

Diseño Experimental:

Se utilizó regresión lineal para modelar la expresión génica tomando en cuenta un diseño factorial 2x2.

Metodología:

El flujo de trabajo comienzá con los datos sin procesar (archivos binarios .CEL, obtenidos del proceso de hibridación). Los pasos seguidos fueron: lectura de datos sin procesar, control de calidad, normalización, filtrado, selección de genes expresados diferencialmente, y comparación de listas seleccionadas. A pesar de que se requiere en un análisis típico, el paso de análisis de significancia biológica no debería incluirse en este informe. (Ver fig. 1)

(Sanz and Sánchez-Pla 2019)

Al guardar el archivo como proyecto (`cml.rproj`), los archivos asociados a al proyecto pueden tratarse utilizando la dirección `./` para referirse al archivo donde se ubica el proyecto. Se crearon 2 carpetas `data` y `results` en el directorio "raíz" del proyecto.

Resultados

El archivo `targets.csv` fue creado de acuerdo con el ejemplo del caso de estudio proporcionado (Sanz and Sánchez-Pla 2020) y tomando en cuenta la guía del usuario de `limma` (Smyth, Thorne, and Wettenhall 2003). A continuación se puede observar el contenido:

Table 1: Contenido del archivo `targets.csv` utilizados para este análisis

file_name	group	control	supplement	short_name
GSM1058965_B324_H_Gene_1.0_ST_1_RB585-NN_.CEL	C.NO	C	NO	C.NO.1
GSM1058966_B325_H_Gene_1.0_ST_2_RB585-NI_.CEL	N.IM	N	IM	N.IM.1
GSM1058967_B326_H_Gene_1.0_ST_3_RB585-SN_.CEL	N.MSC	N	MSC	N.MSC.1
GSM1058968_B327_H_Gene_1.0_ST_4_RB585-SI_.CEL	N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.1
GSM1058969_B328_H_Gene_1.0_ST_5_RB611-NN_.CEL	C.NO	C	NO	C.NO.2
GSM1058970_B329_H_Gene_1.0_ST_6_RB611-NI_.CEL	N.IM	N	IM	N.IM.2
GSM1058971_B330_H_Gene_1.0_ST_7_RB611-SN_.CEL	N.MSC	N	MSC	N.MSC.2

file_name	group	control	supplement	short_name
GSM1058972_B331_H_Gene_1.0_ST_8_RB611-SI_.CEL	N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.2
GSM1058973_B332_H_Gene_1.0_ST_9_RB626-NN_.CEL	C.NO	C	NO	C.NO.3
GSM1058974_B333_H_Gene_1.0_ST_10_RB626-NI_.CEL	N.IM	N	IM	N.IM.3
GSM1058975_B333_H_Gene_1.0_ST_10_RB626-NI_.CEL	N.MSC	N	MSC	N.MSC.3
GSM1058976_B333_H_Gene_1.0_ST_10_RB626-NI_.CEL	N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.3

Lectura de archivos `.CEL`

Para hacer la lectura de los archivos `.CEL` se utilizó las librerías `oligo` y `Biobase`. Además, se debe tomar en cuenta que tienen que instalarse los paquetes requeridos por `R` para ser utilizados. Es recomendable (Sanz and Sánchez-Pla 2020) instalar Rtools si no se está utilizando linux.

Los paquetes utilizados son:

Tabla II. Paquetes de R utilizados en el análisis

`annotate`	`ggplot2`	`knitr`
`arrayQualityMetrics`	`ggrepel`	`limma`
`Biobase`	`ggrepel`	`oligo`
`BiocManager`	`gplots`	`org.Hs.eg.db`
`devtools`	`htmltable`	`pd.hugene.1.0.st.v1`
`genefilter`	`hugene10sttranscriptcluster.db`	`prettydoc`
`printr`	`pvca`	`prettydoc`
`genefilter`	`reactome.db`	`reactomePA`

Es importante notar que algunos paquetes requieren el uso de `BiocManager`, de Bioconductor.

Lectura de los archivos `.CEL`

```
Reading in : ./data/GSM1058965_B324_H_Gene_1.0_ST_1_RB585-NN_.CEL
Reading in : ./data/GSM1058966_B325_H_Gene_1.0_ST_2_RB585-NI_.CEL
Reading in : ./data/GSM1058967_B326_H_Gene_1.0_ST_3_RB585-SN_.CEL
Reading in : ./data/GSM1058968_B327_H_Gene_1.0_ST_4_RB585-SI_.CEL
Reading in : ./data/GSM1058969_B328_H_Gene_1.0_ST_5_RB611-NN_.CEL
Reading in : ./data/GSM1058970_B329_H_Gene_1.0_ST_6_RB611-NI_.CEL
Reading in : ./data/GSM1058971_B330_H_Gene_1.0_ST_7_RB611-SN_.CEL
Reading in : ./data/GSM1058972_B331_H_Gene_1.0_ST_8_RB611-SI_.CEL
Reading in : ./data/GSM1058973_B332_H_Gene_1.0_ST_9_RB626-NN_.CEL
Reading in : ./data/GSM1058974_B333_H_Gene_1.0_ST_10_RB626-NI_.CEL
Reading in : ./data/GSM1058975_B334_H_Gene_1.0_ST_11_RB626-SN_.CEL
Reading in : ./data/GSM1058976_B335_H_Gene_1.0_ST_12_RB626-SI_.CEL
```

Posteriormente, se cambian los nombres de filas y columnas del data frame a nombres más convenientes.

```
GeneFeatureSet (storageMode: lockedEnvironment)
assayData: 1 features, 12 samples
  element names: exprs
protocolData
  rowNames: C.NO.1 N.IM.1 ... N.MSCplusIM.3 (12 total)
  varLabels: exprs dates
  varMetadata: labelDescription channel
phenoData
  rowNames: C.NO.1 N.IM.1 ... N.MSCplusIM.3 (12 total)
```

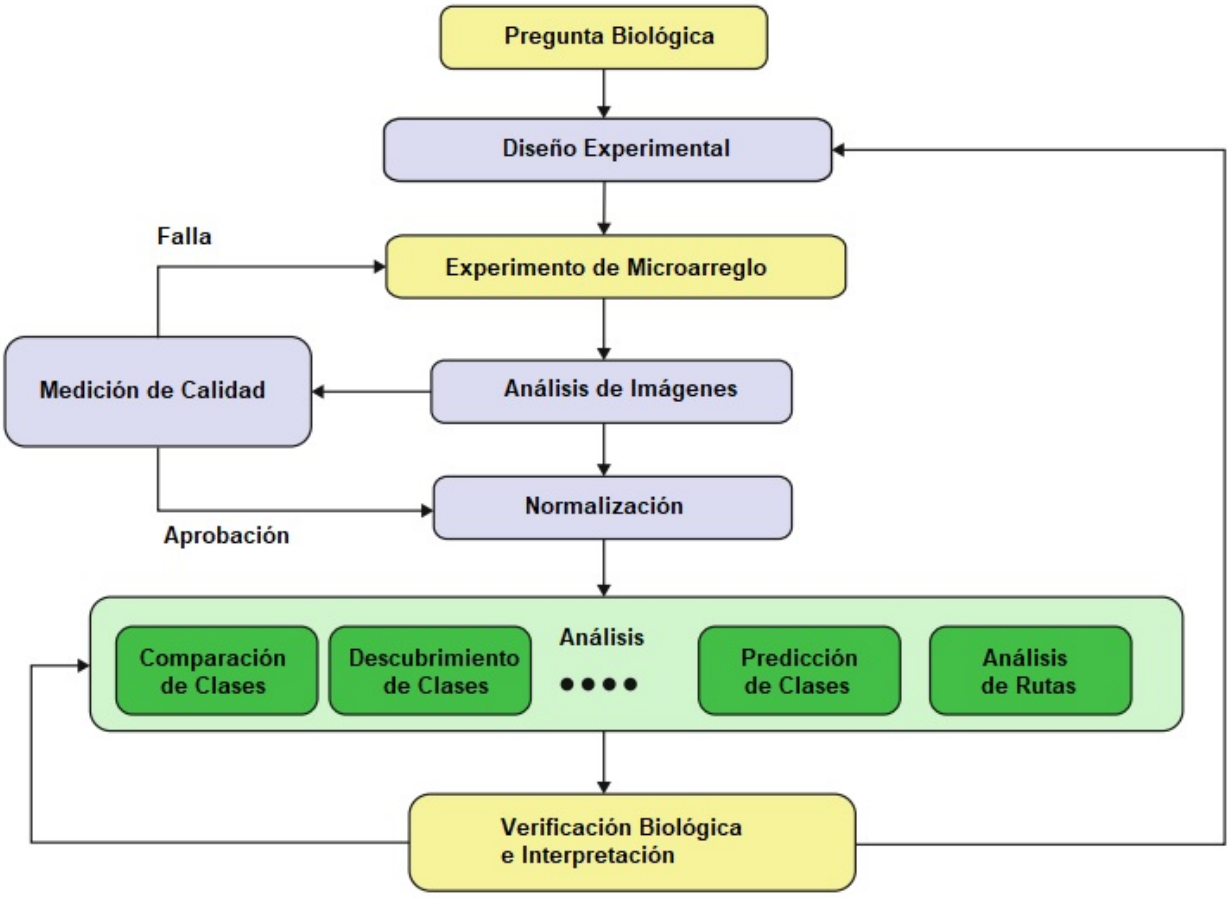

Figure 1: Figura 1. Proceso del análisis de datos de microarreglos

```
  varLabels: group control supplement short_name
  varMetadata: labelDescription channel
featureData: none
experimentData: use 'experimentData(object)'
Annotation: pd.hugene.1.0.st.v1
```

Control de Calidad de los datos no procesados:

Se comprueba si los datos son suficientemente buenos para la normalización. Este es un paso muy relevante, debido a que datos de mala calidad podrían sesgar el análisis, de forma tal que la normalización sería incapaz de arreglar el problema.

Si alguno de los datos `.CEL` se marca tres veces, debería revisarse detalladamente, para revisar si hay algún problema (Ver fig. 2).

	array	sampleNames	*1	*2	*3	group	control	supplement	short_name
☐	1	C.NO.1				C.NO	C	NO	C.NO.1
☐	2	N.IM.1				N.IM	N	IM	N.IM.1
☐	3	N.MSC.1				N.MSC	N	MSC	N.MSC.1
☐	4	N.MSCplusIM.1				N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.1
☐	5	C.NO.2		x		C.NO	C	NO	C.NO.2
☐	6	N.IM.2	x			N.IM	N	IM	N.IM.2
☐	7	N.MSC.2				N.MSC	N	MSC	N.MSC.2
☐	8	N.MSCplusIM.2				N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.2
☐	9	C.NO.3			x	C.NO	C	NO	C.NO.3
☐	10	N.IM.3				N.IM	N	IM	N.IM.3
☐	11	N.MSC.3				N.MSC	N	MSC	N.MSC.3
☐	12	N.MSCplusIM.3				N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.3

Figure 2: Descripción general de metadatos pre-normalización y detección de valores atípicos

Un análsis de componentes principales (PCA) nos perminte disminuir la dimensionalidad de los datos, y disminuir problemas de colinealidad, debido a que se puede orientar la dirección de la variabilidad máxima con los autovectores, y ubicar puntos extremos que se alejan mucho de la distribución del resto de los datos de una forma sencilla. En el análisis realizado no se observaron outliers (Ver fig. 3).

El primer componente del PCA explica 40.6% de la variabilidad total de las muestras. Idealmente se observarían patrones claros para explicar la contribución de las condiciones experimentales a la variabilidad. Pero en este caso, no hay una forma clara de observar dichas contribuciones. Lo único que está claro, es que el comportamiento de la interacción (N.MSCplusIM) parece estar más definido que el de los tratamientos individuales.

Dimensión de los datos sin procesar:

```
[1] 1102500      12
```

Anotación de datos sin procesar:

Indica cuál base de datos debe utilizarse para identificar (anotar) los genes en un momento posterior. Luego de la identificación, la base de datos y la librería (`hugene10sttranscriptcluster.db` y `pd.hugene.1.0.st.v1`), indicadas en la tabla I fueron instaladas de acuerdo con las instrucciones de Bioconductor.

Al parecer, aún sin normalización, los datos sin procesar son bastante homogéneos:

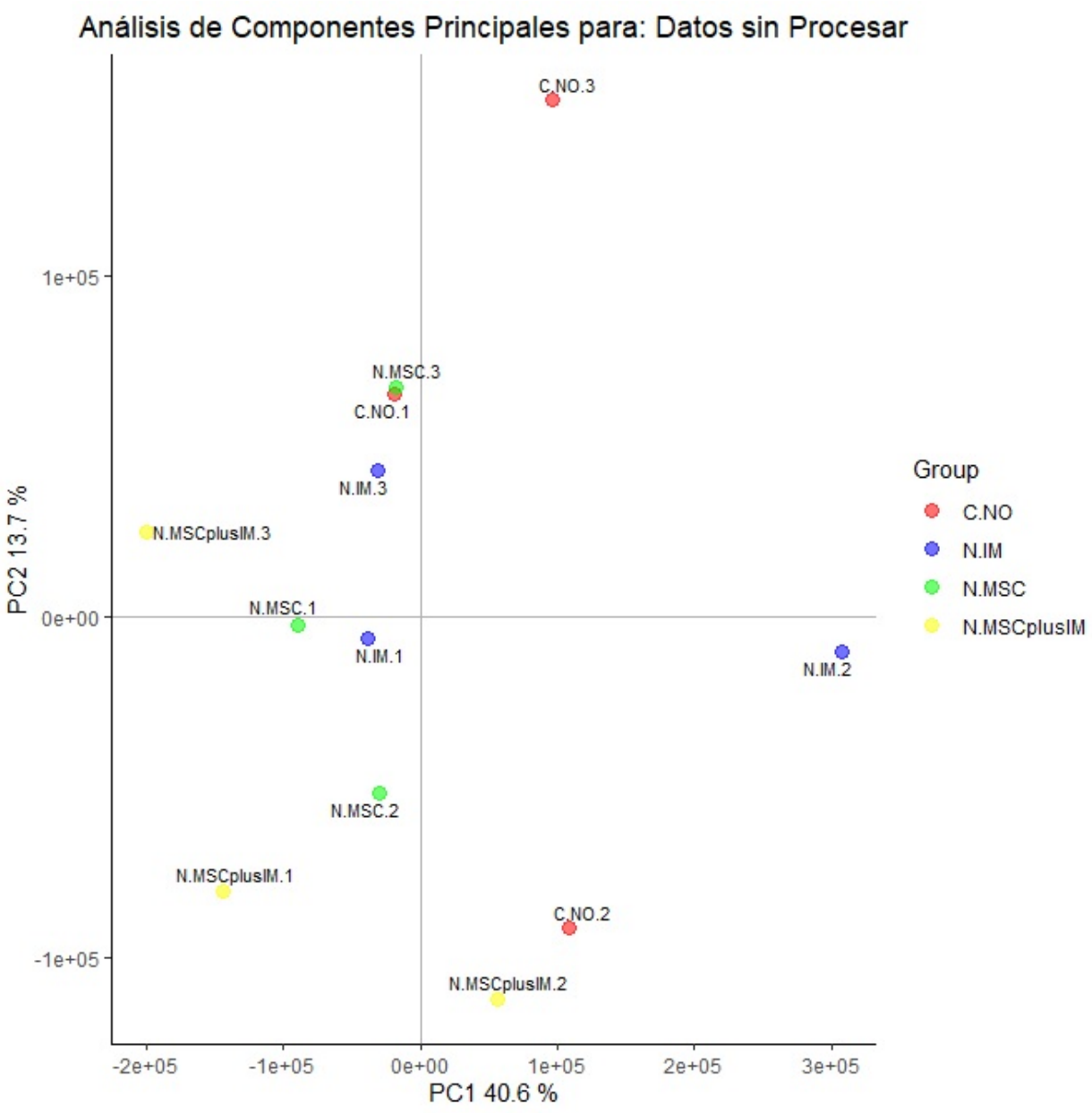

Figure 3: PCA para: Datos sin Procesar

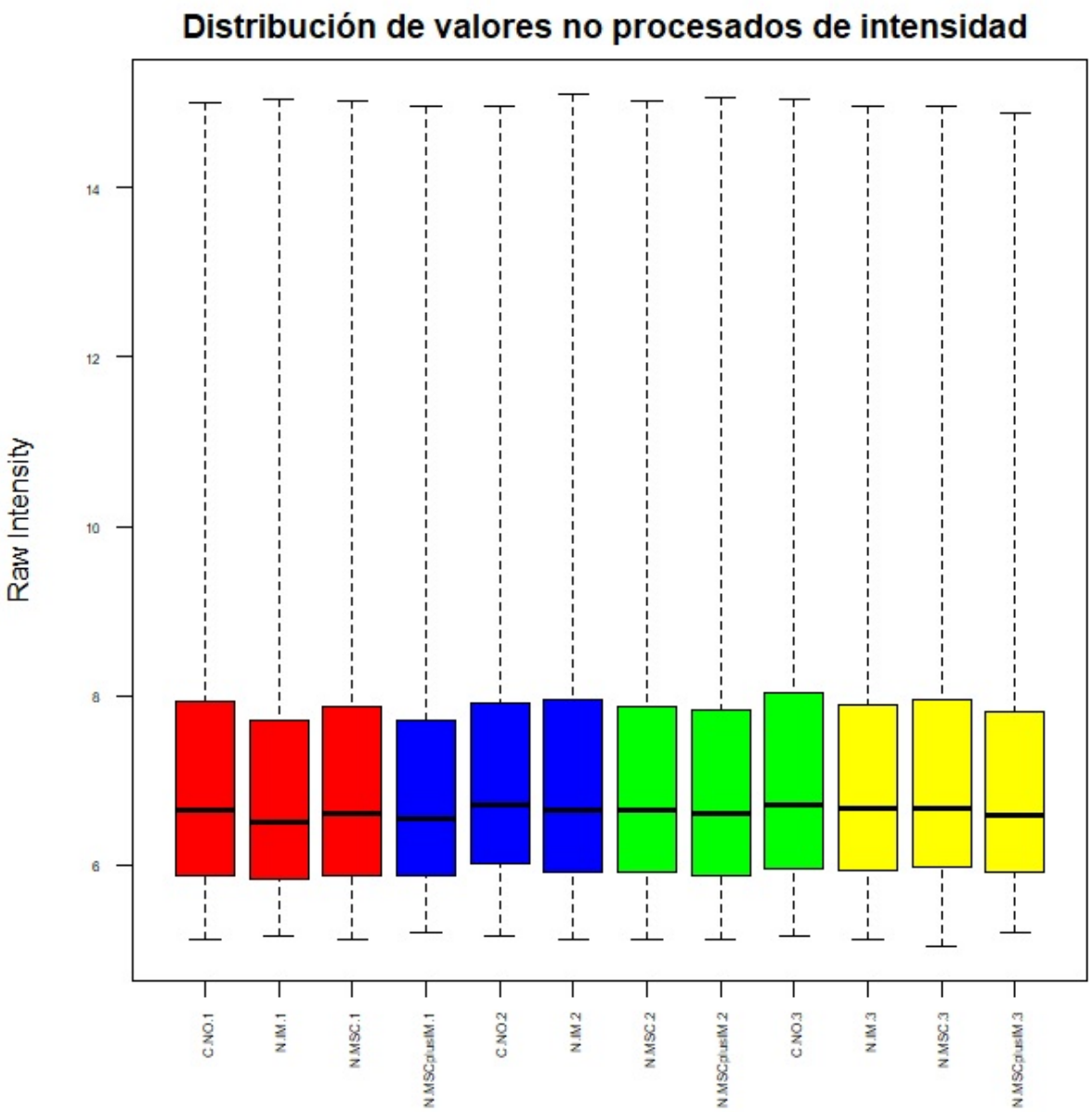

Figure 4: Distribución de valores no procesados de intensidad

Normalización de datos:

Posteriormente se lleva a cabo la normalización de los datos, que consta de 3 pasos. Se realiza con `rma()`:

```
> eset_rma <- rma(raw_data)

Background correcting
Normalizing
Calculating Expression
```

Control de calidad de los datos normalizados:

Nuevamente se revisan los datos para buscar indicios de datos extremos. En este caso no se observan problemas (ver fig. 3).

	array	sampleNames	*1	*2	*3	group	control	supplement	short_name
☐	1	C.NO.1				C.NO	C	NO	C.NO.1
☐	2	N.IM.1				N.IM	N	IM	N.IM.1
☐	3	N.MSC.1				N.MSC	N	MSC	N.MSC.1
☐	4	N.MSCplusIM.1				N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.1
☐	5	C.NO.2				C.NO	C	NO	C.NO.2
☐	6	N.IM.2				N.IM	N	IM	N.IM.2
☐	7	N.MSC.2				N.MSC	N	MSC	N.MSC.2
☐	8	N.MSCplusIM.2				N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.2
☐	9	C.NO.3				C.NO	C	NO	C.NO.3
☐	10	N.IM.3				N.IM	N	IM	N.IM.3
☐	11	N.MSC.3				N.MSC	N	MSC	N.MSC.3
☐	12	N.MSCplusIM.3				N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.3

Figure 5: Descripción general de metadatos normalizados y detección de valores atípicos

Al realizar un PCA con los datos normalizados (ver fig. 6) se puede observar una clara relación entre el tratamiento con MSC y valores más bajos. Al parecer el tratamiento con IM no causa muchos cambios al comparar con los efectos de los controles (C.NO). Sin embargo, el porcentaje de cambios explicado por los datos normalizados es menor (debido a que la disminución de la variabilidad es muy grande al escalar los datos).

Una vez reescalados los datos, se observan pocas diferencias en los mismos. Es importante resaltar que `rma()` incluye un paso ("normalización cuantil"), donde la distribución empírica de todas las muestras se establece en los mismos valores. Como consecuencia, se espera que los diagramas de caja sean idénticos o al menos muy similares (Sanz and Sánchez-Pla 2020) (Ver fig. 7).

Detección de "Errores de Bache"

Se utilizó la librería `pvca` (Principal Component Variation Analysis) para estimar fuente y proporción de variación, a través de análisis de PCA y posteriormente de análisis de varianza de los componentes (Bushel 2013).

Al parecer hay un efecto de bache importante en los datos que tenemos (Ver fig. 8). De existir la posibilidad, se tomaría la opción de investigar en el wet lab si las muestras se procesaron el mismo día. En este caso, no se puede tener la información. Sin embargo, al ser un ejercicio académico, se completará el workflow.

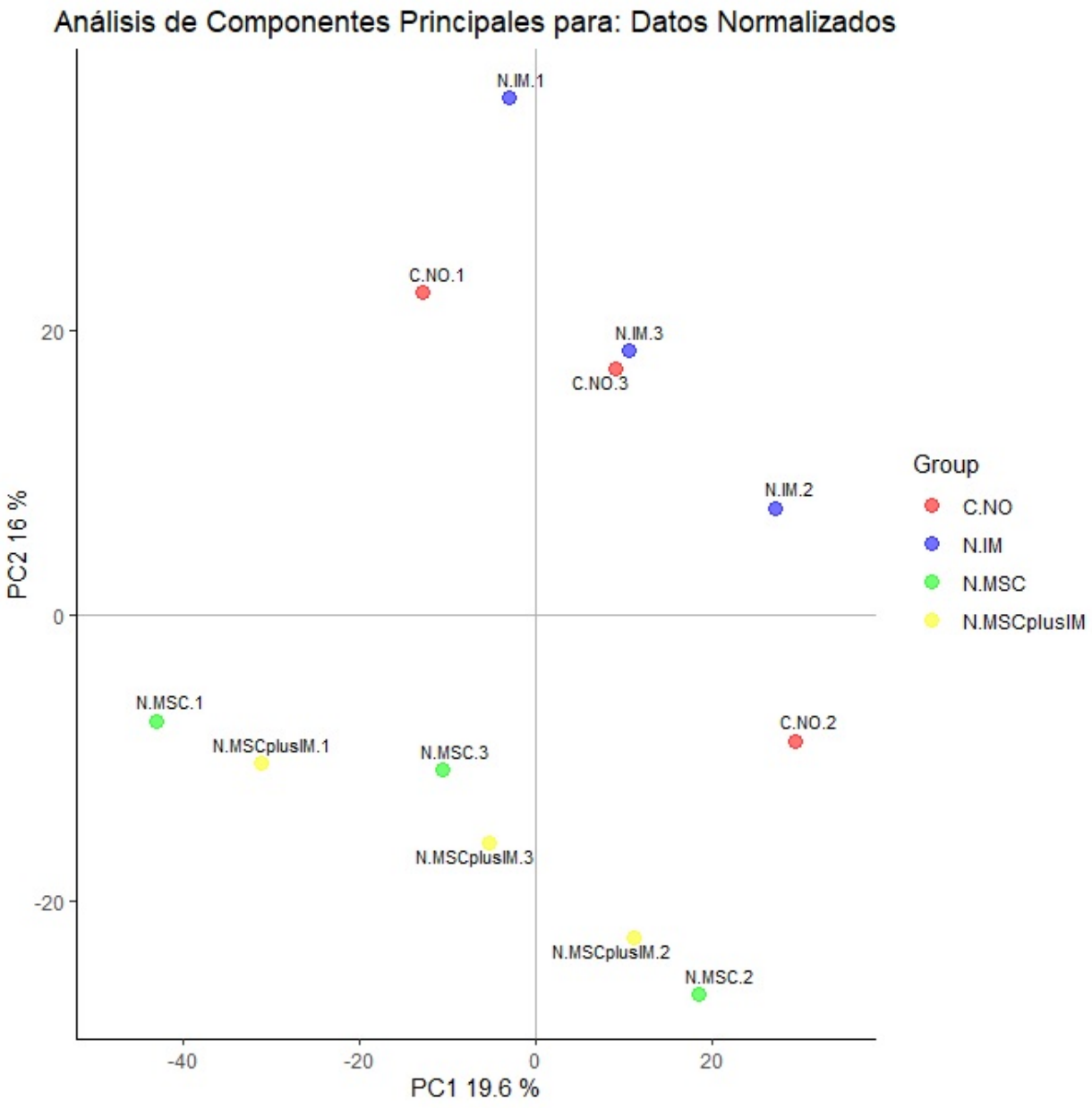

Figure 6: PCA para: Datos Normalizados

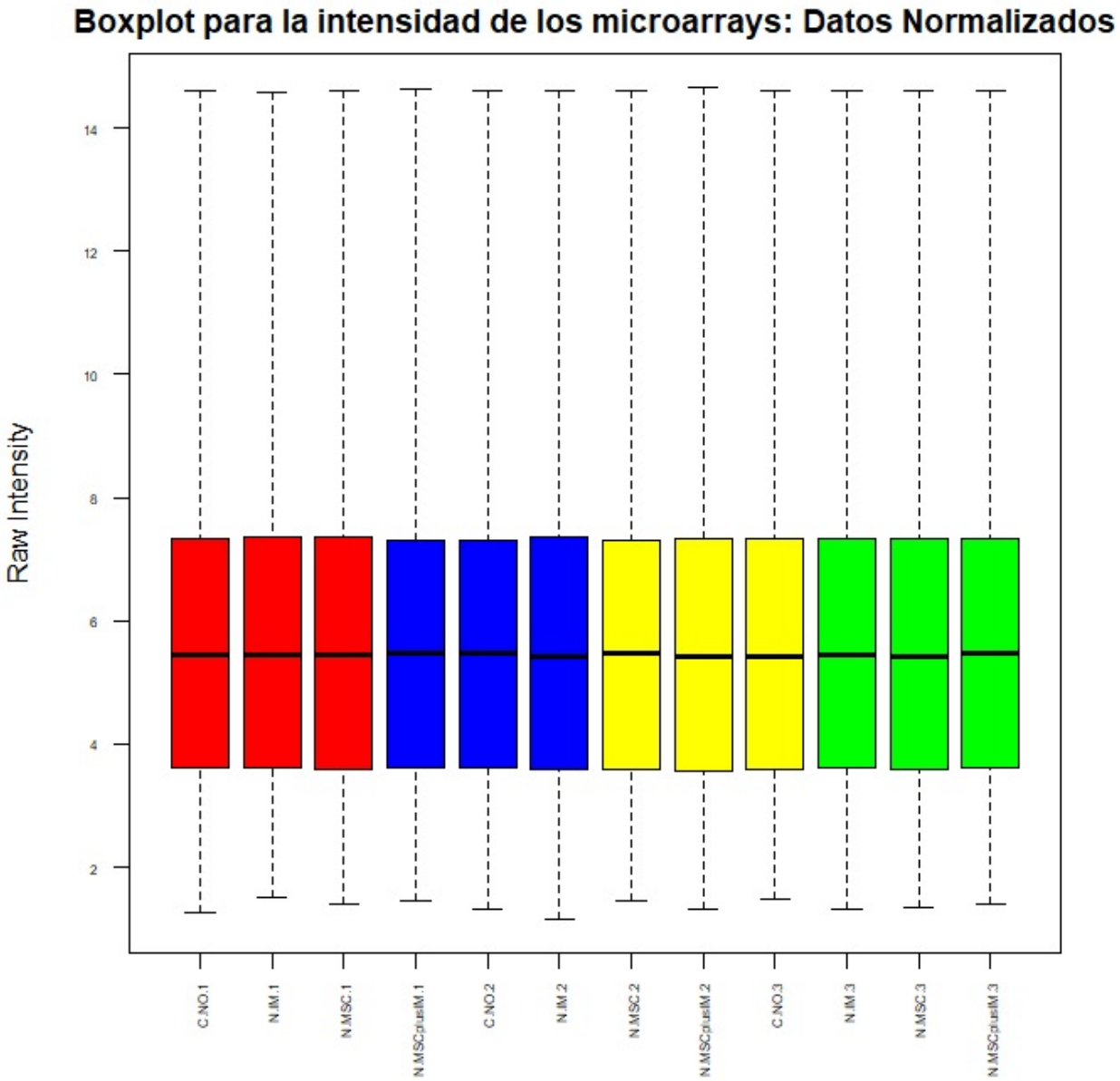

Figure 7: Boxplot para la intensidad de los microarrays: Datos Normalizados

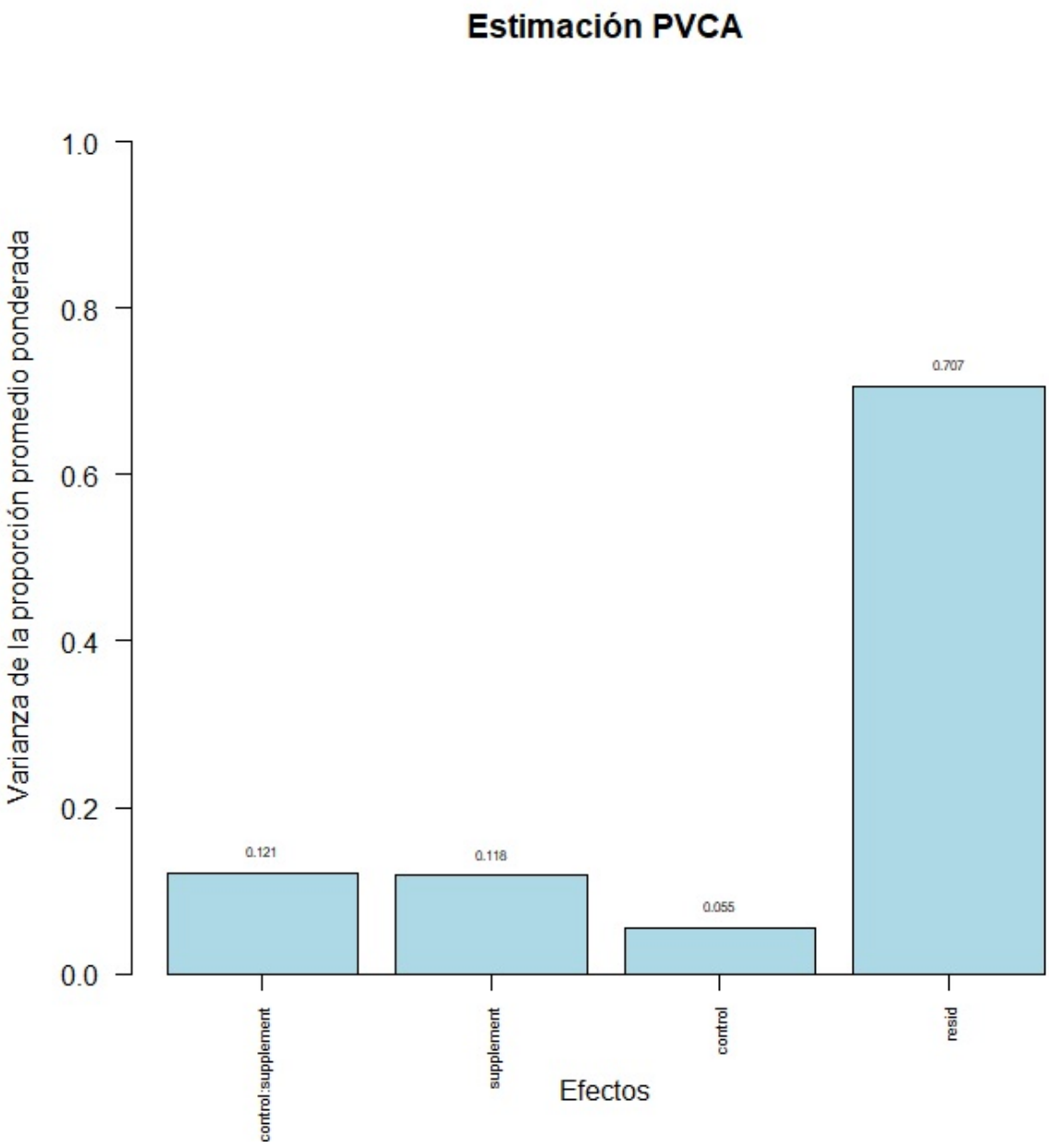

Figure 8: Importancia relativa de los diferentes factores que afectan la expresión génica

Detectando los genes más variables:

La cantidad de genes en que estamos haciendo el estudio afecta la selección, a medida que el número de genes aumenta, más es la necesidad de ajustar los p valores, que de no modificarse causarían un error tipo I demasiado alto.

Si un gen se expresa diferencialmente, esperaríamos que hubiese una diferencia entre los grupos, y que la varianza general del gen sea mayor que la de aquellos que no tienen expresión diferencial. Graficar la variabilidad general de todos los genes nos sirve para decidir qué porcentaje de genes muestra una variabilidad que puede atribuirse a otras causas que no sean variación aleatoria. A continuación se muestran (fig. 9) las desviaciones estándar de todos los genes, ordenadas de menor a mayor. Los genes más variables tienen una desviación por encima de 90-95%.

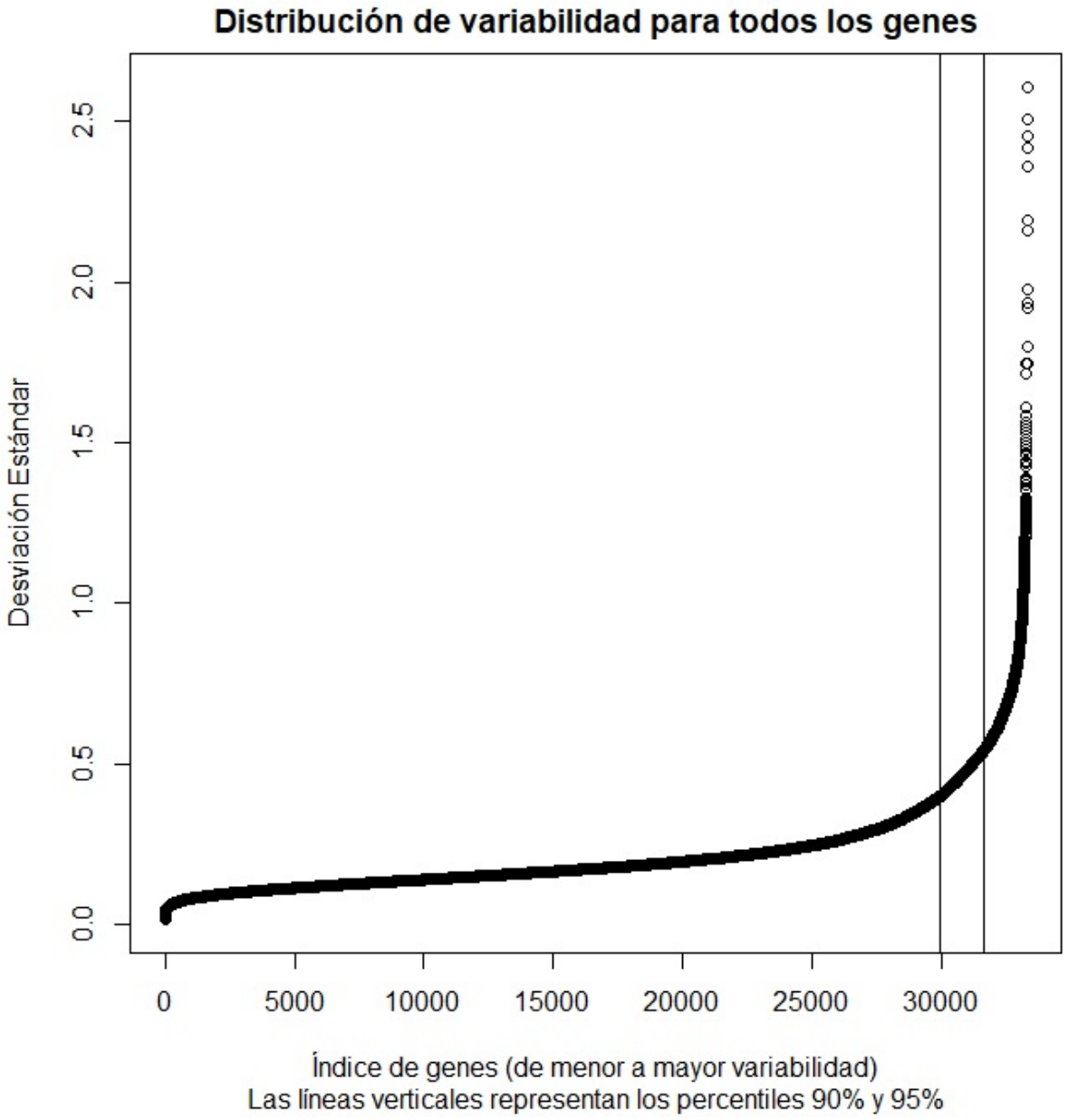

Figure 9: Distribución de variabilidad para todos los genes

Filtrando los genes menos variables:

Filtrar aquellos genes cuya variabilidad se puede atribuir a la variación aleatoria, es decir, los genes que, razonablemente, no se espera que expresen diferencialmente, ha demostrado ser útil para reducir el número de pruebas que se realizarán con el aumento de potencia correspondiente (Sanz and Sánchez-Pla 2020).

La función `nsFilter()` se utilizó para remover genes basándonos en un umbral de variabilidad. Si contamos con un paquete de anotación (que asocia identificadores de sondas de prueba e identificadores de genes de diferentes bases de datos), puede ser utilizado para remover sondas que no tienen un identificador asociado.

A continuación se retorna un reporte de los resultados del filtrado.

```
$numDupsRemoved
[1] 993

$numLowVar
[1] 14116

$numRemoved.ENTREZID
[1] 13483
```

En este caso quedan solamente 4705 genes después del filtrado.

```
Features  Samples
    4705       12
```

Hay diferencias entre los parámetros de este análisis y los del paper revisado: la varianza que utilizan de cutoff es 0.25, y encontraron 12553 genes luego del filtrado. A continuación se muestran los genes relevantes del análisis de Zhang y colaboradores (ver fig. 10, elaborada por (Zhang et al. 2013)).

	array	sampleNames	*1	*2	*3	group	control	supplement	short_name
☐	1	C.NO.1				C.NO	C	NO	C.NO.1
☐	2	N.IM.1				N.IM	N	IM	N.IM.1
☐	3	N.MSC.1				N.MSC	N	MSC	N.MSC.1
☐	4	N.MSCplusIM.1				N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.1
☐	5	C.NO.2				C.NO	C	NO	C.NO.2
☐	6	N.IM.2				N.IM	N	IM	N.IM.2
☐	7	N.MSC.2				N.MSC	N	MSC	N.MSC.2
☐	8	N.MSCplusIM.2				N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.2
☐	9	C.NO.3				C.NO	C	NO	C.NO.3
☐	10	N.IM.3				N.IM	N	IM	N.IM.3
☐	11	N.MSC.3				N.MSC	N	MSC	N.MSC.3
☐	12	N.MSCplusIM.3				N.MSCplusIM	N	MSCplusIM	N.MSCplusIM.3

Figure 10: Figura 10. Ensayos de microarray de expresión genética en células CML CD34+ cocultivadas con y sin MSCs y con y sin IM

(Zhang et al. 2013)

Guardando los datos normalizados:

Es conveniente guardar los datos filtrados normalizados, son el punto de partida para futuros análisis, y es posible que deseemos volver a ellos.

Es habitual guardar los objetos binarios, pero también escribir valores de expresión en archivos de texto o Excel. Escribir en Excel desde R no es una tarea trivial, por extraño que parezca, porque los diferentes paquetes funcionan de manera diferente según el sistema operativo (Sanz and Sánchez-Pla 2020). Una alternativa puede ser cortar y copiar la información de interés a mano; es importante mencionar que al no utilizar código, el error humano es un factor que debe considerarse.

```
> write.csv(exprs(eset_rma), file = "./results/normalized.Data.csv")
> write.csv(exprs(eset_filtered), file = "./results/normalized.Filtered.Data.csv")
> save(eset_rma, eset_filtered, file = "./results/normalized.Data.Rda")
```

Definición de la configuración experimental: la matriz de diseño

```
   C.NO N.MSC N.MSCplusIM N.IM
1     1     0           0    0
2     0     1           0    0
3     0     0           1    0
4     0     0           0    1
5     1     0           0    0
6     0     1           0    0
7     0     0           1    0
8     0     0           0    1
9     1     0           0    0
10    0     1           0    0
11    0     0           1    0
12    0     0           0    1
attr(,"assign")
[1] 1 1 1 1
attr(,"contrasts")
attr(,"contrasts")$group
[1] "contr.treatment"
```

Definición de la configuración experimental: la matriz de contraste

```
            Contrasts
Levels        IM CMR INT
  C.NO         1   1   1
  N.MSC        0  -1   0
  N.MSCplusIM  0   0  -1
  N.IM        -1   0   0
```

```
[1] "MArrayLM"
attr(,"package")
[1] "limma"
```

Obtención de listas de genes expresados diferencialmente:

El paquete `limma` implmenta la función `topTable()`, que contiene una lista de genes ordenados de menor a mayor p valor, que puede ser considerada de los genes más a menos diferencialmente expresados (Sanz and Sánchez-Pla 2020).

Los parámetros estadísticos indicados por la función son: * `logFC`: diferencia media entre grupos. * `AveExpr`: expresión promedio de todos los genes en la comparación. * `t`: estadístico t moderado (estadístico similar a la prueba t para comparación). * `P.value`: P valor. * `adj.P.Val`: valor p ajustado según Benjamini y Hochberg (1995) * `B`: estadístico B, es decir, la posibilidad de registro posterior del gen de ser vs no ser diferencial expresado (efecto estadístico).

A continuación se muestran los primeros 6 valores de los genes seleccionados con cada interacción. Recordemos que IM contrasta IM vs Control, CMR contrasta CMR vs Control e INT contrasta IM+CMR vs Control.

`topTable(IM)`

```
           logFC  AveExpr         t      P.Value  adj.P.Val        B
7909164 -2.433725 7.276534 -7.983475 3.143688e-06 0.01479105 4.434698
7917561 -1.922731 5.603632 -6.534037 2.406183e-05 0.04499708 2.776849
7948213  2.703593 7.271281  6.245978 3.724809e-05 0.04499708 2.406787
8029530 -1.094250 7.919329 -6.167932 4.201171e-05 0.04499708 2.304094
8118158  1.066664 8.331110  6.084578 4.781836e-05 0.04499708 2.193265
7903786 -1.525071 8.572948 -5.873848 6.662021e-05 0.05224135 1.907749
```

`topTable(CMR)`

```
           logFC   AveExpr        t      P.Value  adj.P.Val          B
7904726 1.8181881 10.673241 7.124793 1.016671e-05 0.04783438  1.1061850
8162039 0.7915146  6.814215 5.426726 1.374380e-04 0.32332298 -0.1178162
8095343 1.2848950  6.542369 4.620919 5.436009e-04 0.42192518 -0.8693532
8118149 0.7395021  7.981456 4.612580 5.516419e-04 0.42192518 -0.8777505
7986350 1.1457020  6.792968 4.588173 5.758949e-04 0.42192518 -0.9023996
8118158 0.8032365  8.331110 4.581908 5.822973e-04 0.42192518 -0.9087439
```

`topTable(INT)`

```
           logFC   AveExpr         t      P.Value   adj.P.Val         B
7909164 -2.583213  7.276534 -8.473848 1.671733e-06 0.007865503 4.5677078
8094240 -1.445042  8.408255 -7.186730 9.313059e-06 0.021908971 3.3174128
7948213  2.643206  7.271281  6.106468 4.621558e-05 0.072481438 2.0652254
7904726 -1.412914 10.673241 -5.536678 1.147198e-04 0.113850068 1.3230120
8157446 -1.067983  3.288384 -5.504195 1.209884e-04 0.113850068 1.2789560
8051583  2.361188  5.716562  5.174808 2.092459e-04 0.141458917 0.8216422
```

Anotación de Genes:

Una vez que se tiene la tabla, debe utilizarse de modo que nos sirva, y para poder entender cuáles genes tenemos necesitamos asociar los identificadores que aparecen en la tabla, generalmente correspondientes a sondas o transcripciones según el tipo de arreglo, con nombres más familiares, como el símbolo del gen, el identificador del gen de Entrez o la descripción del gen: Este proceso es la anotación.

```
> # Gene Annotation function
> annotatedTopTable <- function(topTab, anotPackage) {
+     topTab <- cbind(PROBEID = rownames(topTab), topTab)
+     myProbes <- rownames(topTab)
+     thePackage <- eval(parse(text = anotPackage))
+     geneAnots <- select(thePackage, myProbes, c("SYMBOL", "ENTREZID",
+         "GENENAME"))
+     annotatedTopTab <- merge(x = geneAnots, y = topTab, by.x = "PROBEID",
+         by.y = "PROBEID")
+     return(annotatedTopTab)
+ }
```

Esta función permite la asociación de el paquete de anotación (`hugene10sttranscriptcluster.db` en este caso) con las salidas de `topTabs()` y retornar la información en un formato inteligible. Nótese que se incluye el símbolo, la ID de Entrez y el nombre del gen entre los datos de anotación.

Una vez anotada, la tabla es más comprensible. Nótese el ejemplo bajo éstas líneas, donde se muestran las primeras 4 columnas del registro creado para IM. También podemos relacionar el ProbeID con el Gen.

```
  PROBEID    SYMBOL ENTREZID                                    GENENAME
1 7896759 LINC01128   643837 long intergenic non-protein coding RNA 1128
2 7896798   PLEKHN1    84069    pleckstrin homology domain containing N1
3 7896817     ISG15     9636               ISG15 ubiquitin like modifier
4 7896822      AGRN   375790                                      agrin
5 7896861   MIR200A   406983                                microRNA 200a
```

Visualización de genes expresados diferencialmente:

A continación se mostrarán "Volcano Plots": son un tipo de diagrama de dispersión que muestra significancia estadística (valor P) versus magnitud de cambio (Fold Change). Permite la identificación visual rápida de genes con grandes cambios de expresión que también son estadísticamente significativos. Es muy posible que sean los genes biológicamente más significativos. En las figuras a continuación se muestran los genes con menor P valor y diferencialmente expresados para cada contraste realizado, en cada mapa se destacan los genes con menor $-log_{10}(pvalor)$ y $log_2|FC| \geq 1$ (Ver figs 11, 12, 13).

Comparaciones múltiples:

Generalmente, es interesante saber cuáles genes se han seleccionado en cada comparación. en este caso, los genes biológicamente relevantes son aquellos que se seleccionaron en cada contraste especifico.

```
          IM  CMR  INT
Down      10    0    2
NotSig 4693 4704 4702
Up        2    1    1
```

A continuación, se muestra un diagrama de Venn que representa el número de genes expresados diferencialmente en cada comparación, con un punto de corte dado ($-log_{10}(pvalor)$ y $log_2|FC| \geq 1$). Es importante destacar que debido al diseño experimental, El tratamiento con la interacción estaba en 3 viales, lo que causa que la intersección de los 3 tratamientos sea 0.

Heatmaps:

Este tipo de gráfico nos permite ver los niveles de expresión de distintos genes en diferentes tratamientos. En este análsis mostraremos sólamente 1 heatmap considerando los genes agrupados se puede notar que en general, los tratamientos con IM y los Controles dan resultados más bajos de expresión, con la excepción del gen PRG3, y AIF1 en caso de los controles y el tratamiento combinado. (Ver fig. 15).

Significado biológico de los resultados

La cantidad de genes de interés por tratamiento se resume a continuación.

```
 IM CMR INT
 79   1  10
```

Discusión

Las limitaciones principales del análisis fueron:

- Falta de información de la fecha de los análisis para determinar si hubo batch effect.
- Los resultados obtenidos fueron distintos a los del paper. También es cierto que desde 2013 se han actualizado las bases de datos sobre genes humanos y que lo más posible es que de utilizarse el mismo cutoff se obtendría un numero mayor. Sería interesante investigar si la cantidad de genes únicos se mantiene.

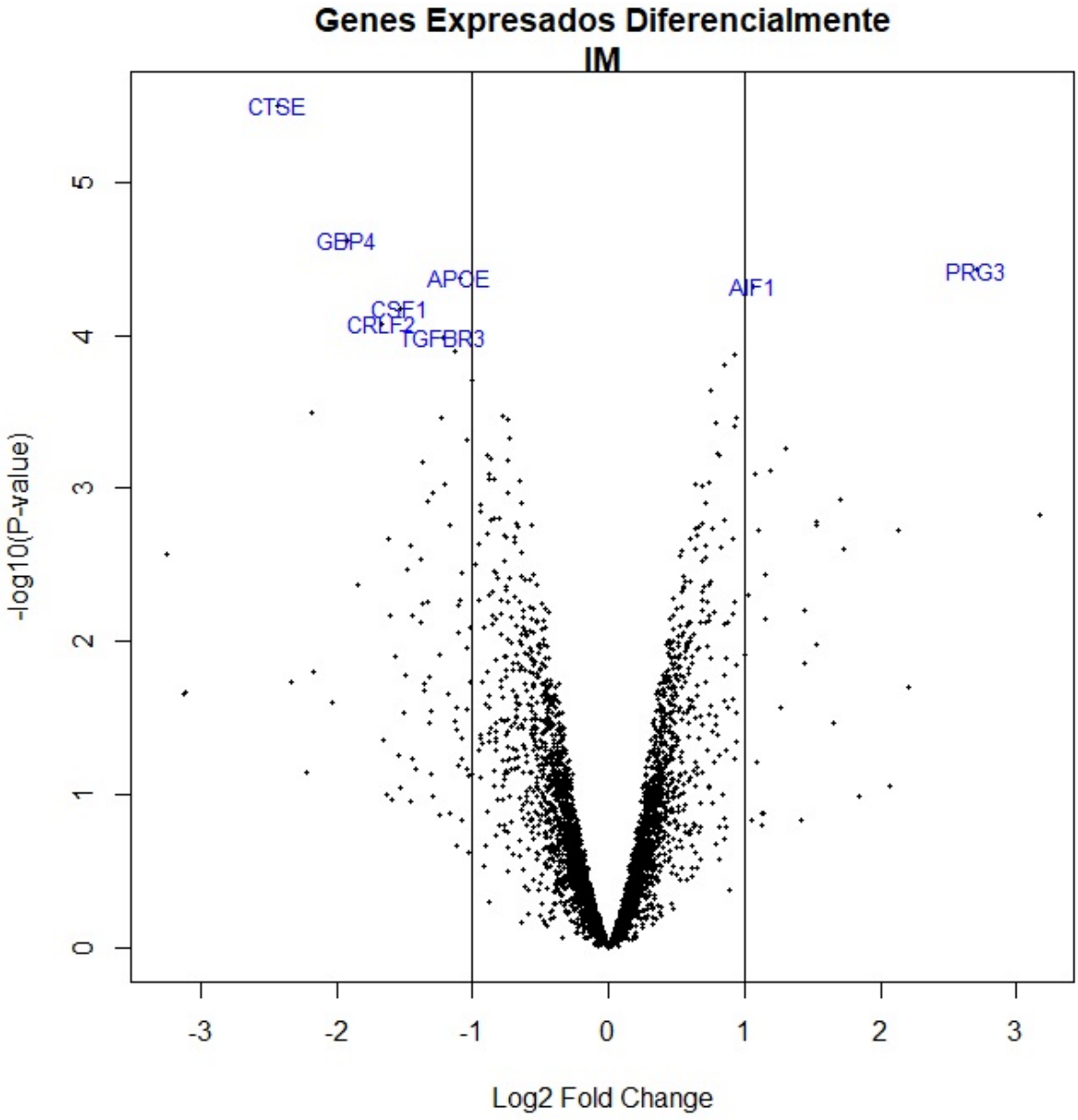

Figure 11: Volcano plot para IM

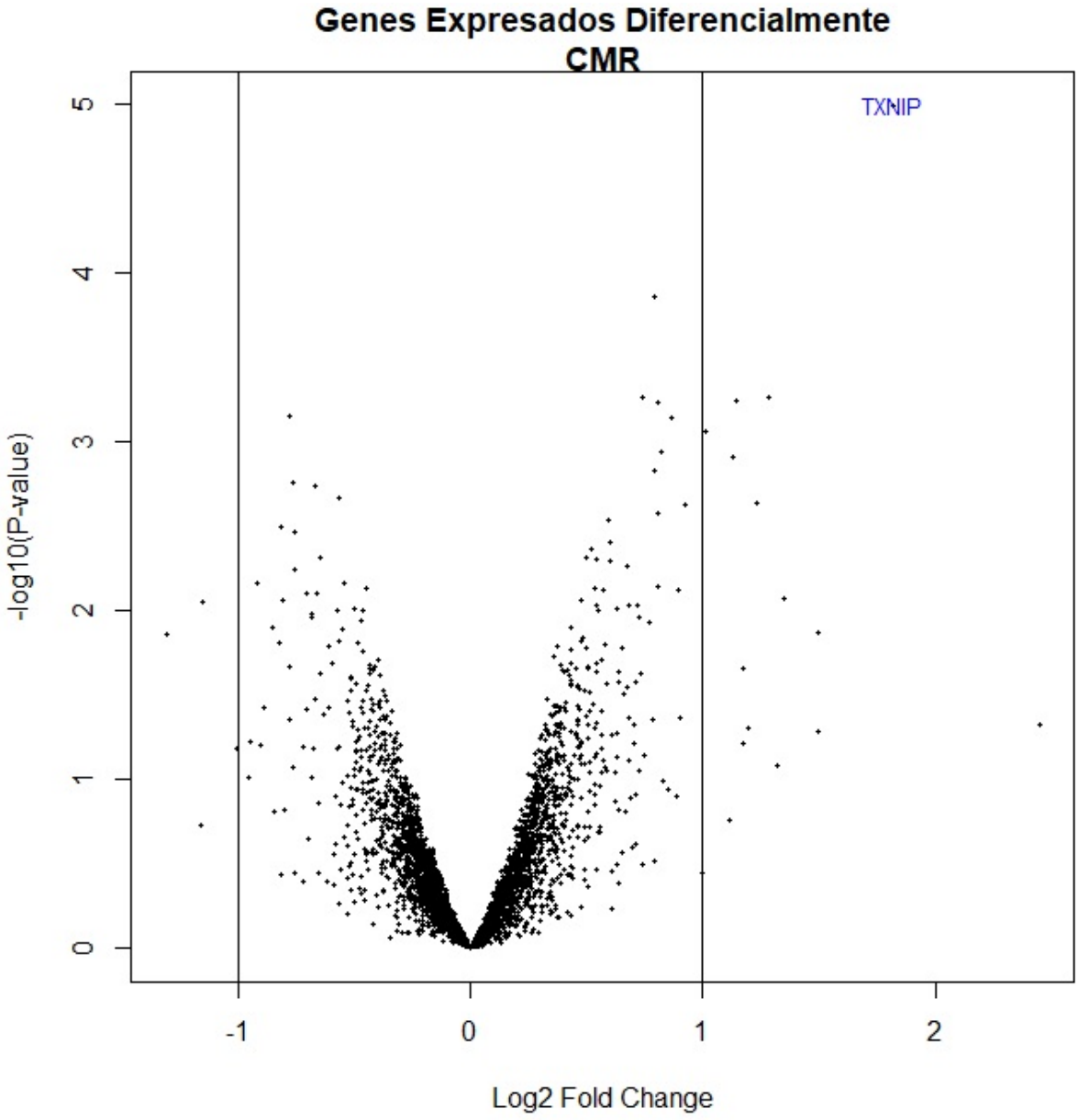

Figure 12: Volcano plot para CMR

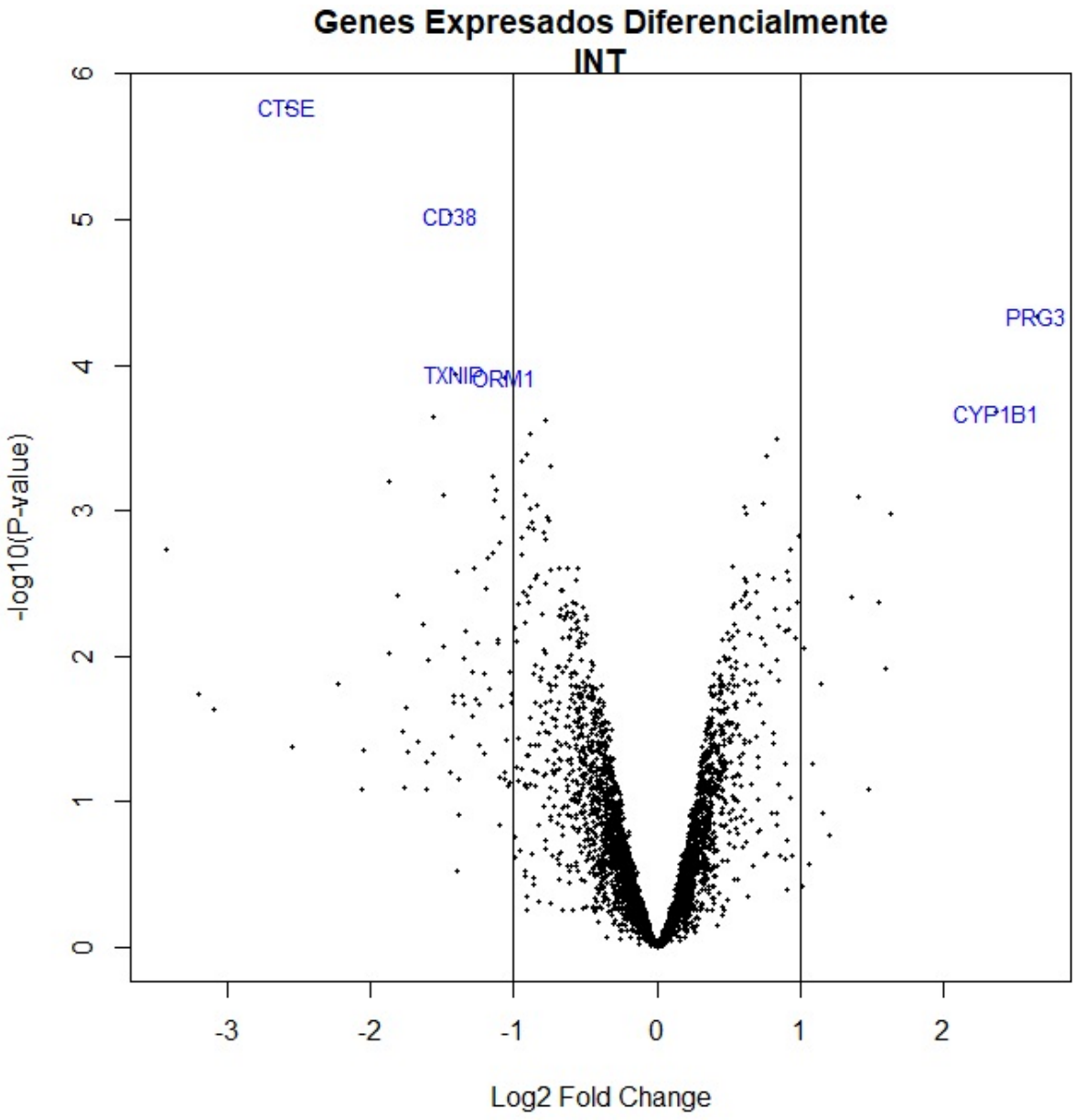

Figure 13: Volcano plot para CMR+IM (INT)

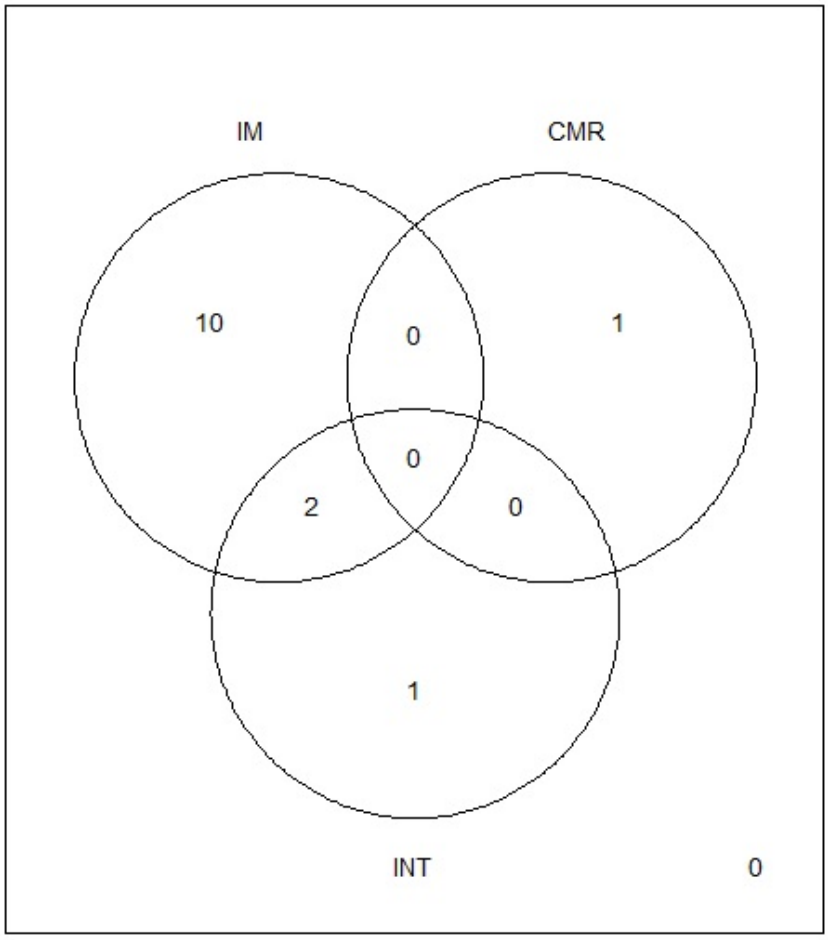

Figure 14: Genes en común entre las 3 comparaciones

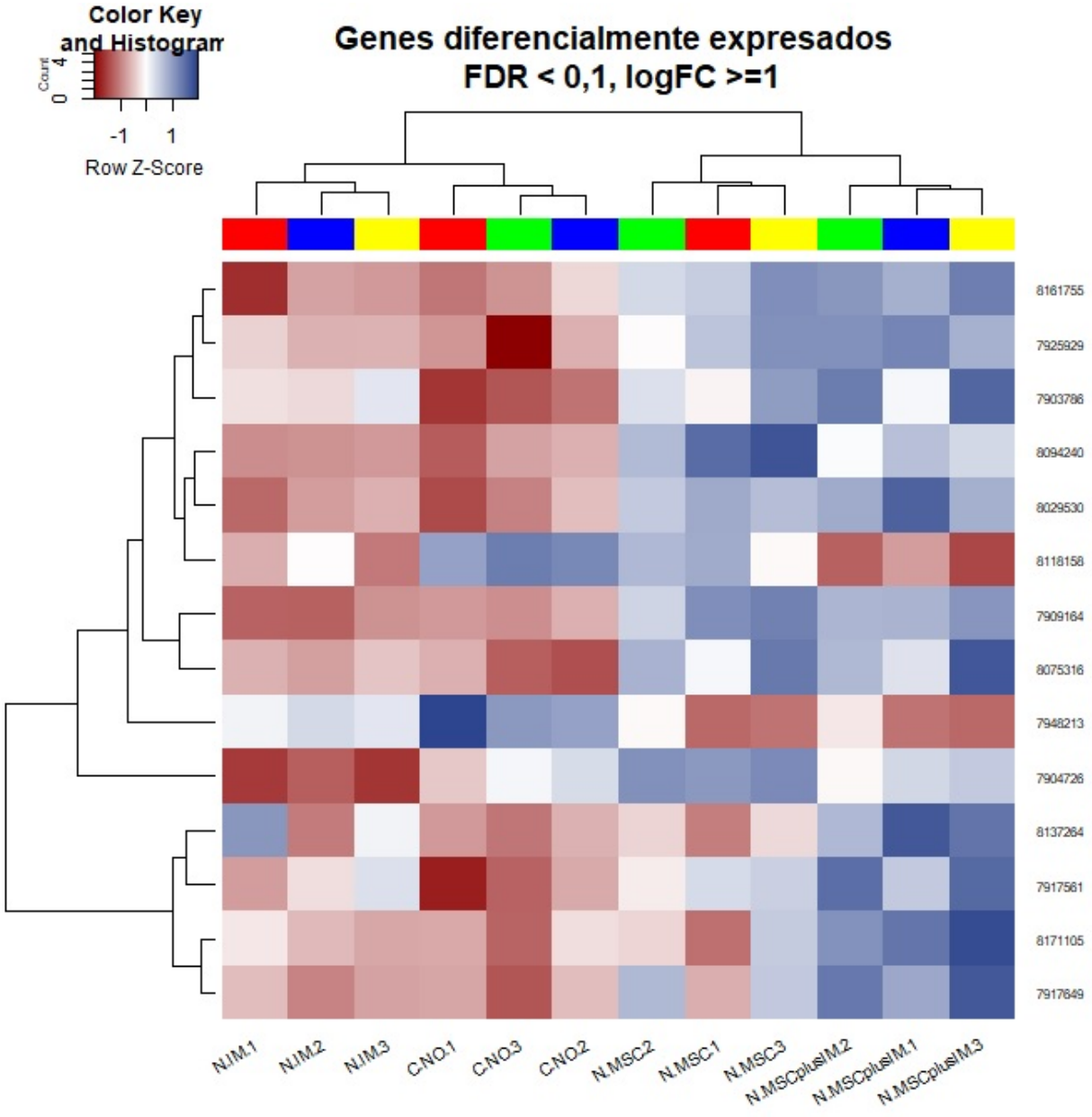

Figure 15: Genes diferencialmente expresados, agrupados

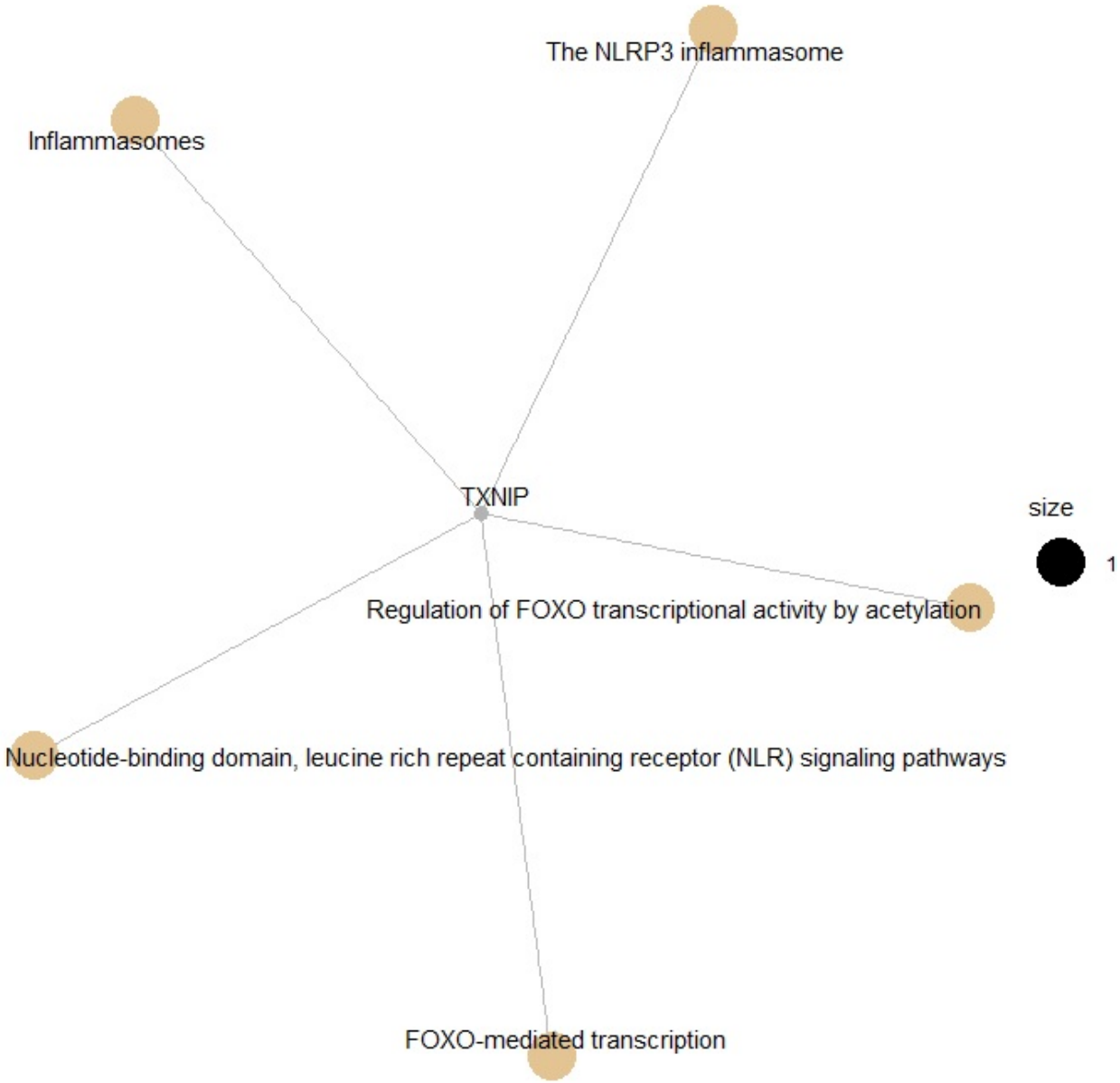

Figure 16: Mapa de red de interacción con CMR

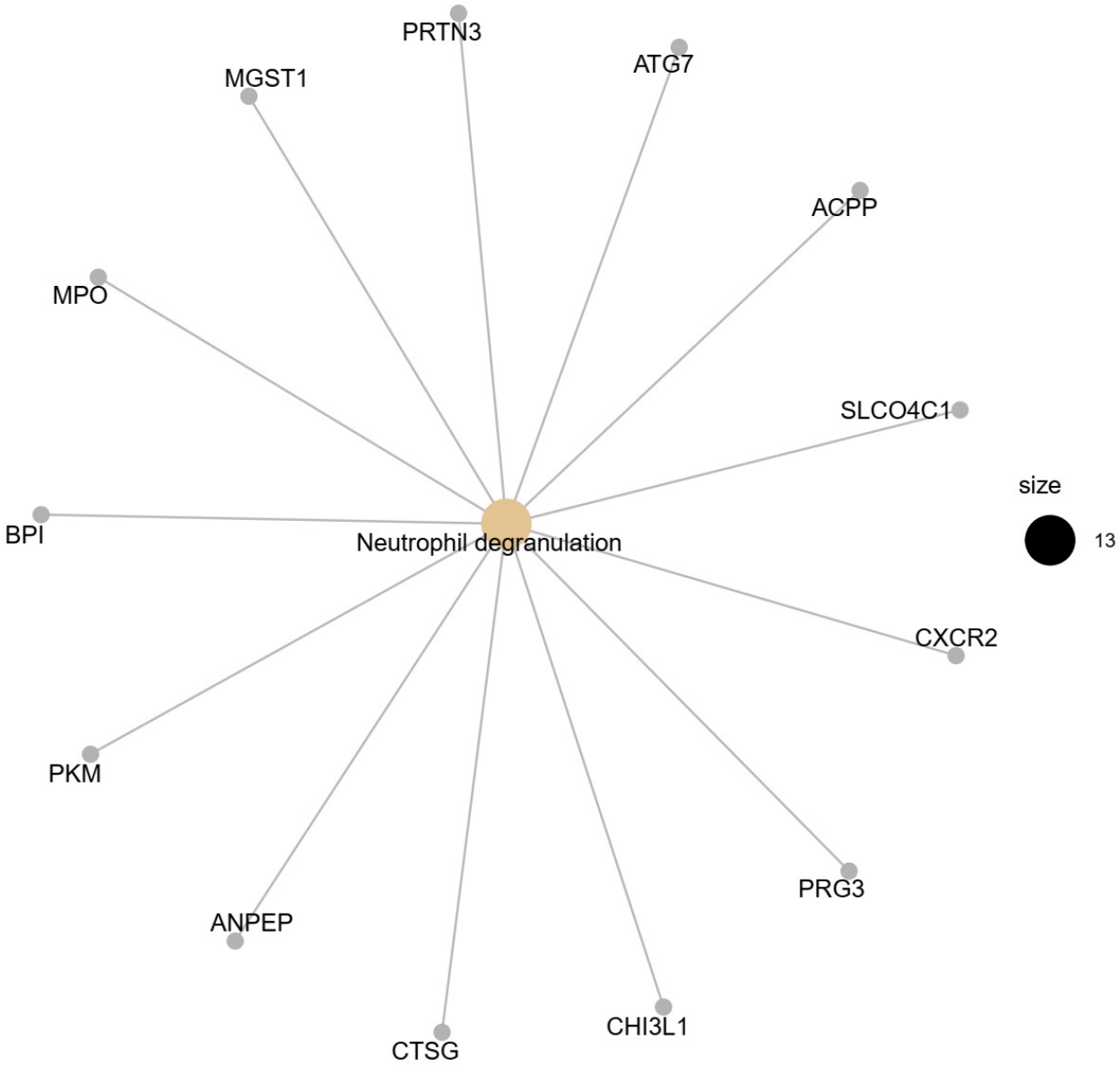

Figure 17: Mapa de red de interacción con IM

- En general, se utilizó el formato: `if(!(require(package))) install.packages("package")` o `if(!(require(package))) BiocManager::install("package")` para instalar paquetes, pero en el caso de las bases de datos, no se realizó la instalación con ese comando, tuvo que utilizarse la instalación directa.
- No encontré forma de obtener un mapa de la interacción (CMR+IM), sin embargo queda un registro de los genes involucrados. Tomando en cuenta que CMR tenía sólo un gen, es posible que la intersección diese vacío.

Conclusión

- Esta parte del trabajo no es requerimiento.
- A pesar de que R viene sin garantías, y que a veces puede haber falta de compatiblidad, la documentación es muy completa, y casi cualquier duda puede solventarse a través de la consulta de páginas web: documentación R - documentación Bioconductor y foros en línea. La presencia de una comunidad muy extensa es ventajosa a la hora de buscar cómo solventar un problema.

Referencias

Bushel, P. 2013. "Pvca: Principal Variance Component Analysis (Pvca)." *R Package Version* 1 (0).

Sanz, Ricardo Gonzalo, and Alex Sánchez-Pla. 2019. "Statistical Analysis of Microarray Data." In *Microarray Bioinformatics*, 87–121. Springer.

———. 2020. "Statistical Analysis of Microarray Data."

Smyth, Gordon K, Natalie Thorne, and James Wettenhall. 2003. "Limma: Linear Models for Microarray Data User's Guide." *Software Manual Available from Http://Www. Bioconductor. Org.*

Zhang, Bin, Min Li, Tinisha McDonald, Tessa L Holyoake, Randall T Moon, Dario Campana, Leonard Shultz, and Ravi Bhatia. 2013. "Microenvironmental Protection of Cml Stem and Progenitor Cells from Tyrosine Kinase Inhibitors Through N-Cadherin and Wnt–β-Catenin Signaling." *Blood, the Journal of the American Society of Hematology* 121 (10): 1824–38.