ballester_gabriel_ADO_PEC1

Gabriel Ballester Lozano

4/21/2020

Estudio escogido

El estudio que he escogido para realizar la presente PEC es el estudio https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE133975, pubicado el 17 de abril de 2020 y titulado "Reprogrammed alveolar macrophages after pneumonia recovery". El organismo de estudio es Mus musculus y el diseño del estudio es comparativo

Obtenemos los datos del estudio, aunque dada la extensión de la lista de phenodata, preferimos realizar nuestra propio archivo de targets para simplificarlo:

```
> #elist <- getGEO("GSE133975")
> celFiles <- list.celfiles("./data", full.names = TRUE)
> my.targets <-read.AnnotatedDataFrame(file.path("./data","targets.csv"),
+                                      header = TRUE, row.names = 1,
+                                      sep=";")
> rawData <- read.celfiles(celFiles, phenoData = my.targets)
> my.targets@data$ShortName->rownames(pData(rawData))
> colnames(rawData) <-rownames(pData(rawData))
```

En el estudio usaron microarrays para detallar la expresión génica de macrófagos alveolares de ratones infectados con la bacteria que causa la neumonía denominada pneumococcus -aunque en la id de las muestra contendrá el valor de SP3-, que se dividen en dos grupos (cada uno con su control); ratones "naive" y ratones ya infectados y recuperados pasado un més. A ambos ratones se les tomó una muestra de pulmón para el análisis de macrófagos, que son las células objetivo de este estudio.

La plataforma de Affymetrix utilizada para este estudio es [MoGene-2_0-st] Affymetrix Mouse Gene 2.0 ST Array mogene20st_Mm_ENTREZG_17.1.0 basada en oligonucleotidos in situ. Por ello, cabe esperar que el paquete de anotaciones de bases de datos de bioconductor "mogene21sttranscriptcluster.db" sea el adecuado para las anotaciones de los genes correspondientes.

El presente archivo y sus resultados se hallan disponibles en el repositorio de Github https://github.com/GABRIELBALLESTER/Ballester_Gabriel_ADO_PEC1.git

Control de calidad

Al crear el directorio "arrayQualityMetrics_report_for_rawData" observamos el archivo index.html y vemos que tan solo hay una marca en algunos arrays por lo que se puede decidir seguir adelante con todos;

```
> #arrayQualityMetrics(rawData)
```

Sin embargo, para un control de calidad de los datos brutos más exhaustivo, procederemos con el siguiente análisis de componentes principales o PCA;

```
> plotPCA3 <- function (datos, labels, factor, title, scale,colores, size = 1.5, glineas = 0.25) {
+   data <- prcomp(t(datos),scale=scale)
+   # plot adjustments
```

- Array metadata and outlier detection overview

	array	sampleNames	*1	*2	*3	Group	Experimental	Ce
☐	1	N.C.1				N.C	Control	
☐	2	N.C.2				N.C	Control	
☑	3	N.C.3			x	N.C	Control	
☐	4	N.C.4				N.C	Control	
☐	5	N.SP3.1				N.SP3	SP3	
☐	6	N.SP3.2				N.SP3	SP3	
☐	7	N.SP3.3				N.SP3	SP3	
☑	8	N.SP3.4			x	N.SP3	SP3	
☐	9	E.C.1				E.C	Control	Exper
☐	10	E.C.2				E.C	Control	Exper
☑	11	E.C.3	x			E.C	Control	Exper
☑	12	E.C.4			x	E.C	Control	Exper
☐	13	E.SP3.1				E.SP3	SP3	Exper
☑	14	E.SP3.2		x		E.SP3	SP3	Exper
☑	15	E.SP3.3			x	E.SP3	SP3	Exper
☑	16	E.SP3.4			x	E.SP3	SP3	Exper

The columns named *1, *2, ... indicate the calls from the different outlier detection methods

1. outlier detection by Distances between arrays
2. outlier detection by Boxplots
3. outlier detection by MA plots

Figure 1: Tabla resumen de los datos del estudio

```
+   dataDf <- data.frame(data$x)
+   Group <- factor
+   loads <- round(data$sdev^2/sum(data$sdev^2)*100,1)
+   # main plot
+   p1 <- ggplot(dataDf,aes(x=PC1, y=PC2)) +
+     theme_classic() +
+     geom_hline(yintercept = 0, color = "gray70") +
+     geom_vline(xintercept = 0, color = "gray70") +
+     geom_point(aes(color = Group), alpha = 0.55, size = 3) +
+     coord_cartesian(xlim = c(min(data$x[,1])-5,max(data$x[,1])+5)) +
+     scale_fill_discrete(name = "Group")
+   # avoiding labels superposition
+   p1 + geom_text_repel(aes(y = PC2 + 0.25, label = labels),segment.size = 0.25, size = size) +
+     labs(x = c(paste("PC1",loads[1],"%")),y=c(paste("PC2",loads[2],"%"))) +
+     ggtitle(paste("Principal Component Analysis for: ",title,sep=" "))+
+     theme(plot.title = element_text(hjust = 0.5)) +
+     scale_color_manual(values=colores)
+   }
```

```
> plotPCA3(exprs(rawData), labels = rawData@phenoData@data[["ShortName"]], factor = rawData@phenoData@da
+          title="Raw data", scale = FALSE, size = 3,
+          colores = c("red", "blue", "green", "yellow"))
```

Tras ver estos resultados he comprobado que los etiquedados están correctamente, puesto que hay dos Arrays que se separan bastante del resto (E.C.3 y E.SP3.4), ambos pertenecientes al tipo "experienced" pero no del mismo grupo experimental. Estos resultados tambien se encuentran en la carpeta "arrayQualityMetrics_report_for_rawData". Quizá deberían eliminarse de los posteriores análisis puesto que estos dos Arrays se encuentran a ambos lados del eje que explica el 49.6% de la variabilidad, no obstante continúo teniéndolos en cuenta y efectuamos el análisis de boxplot;

```
> boxplot(rawData, cex.axis=0.5, las=2,  which="all",
+          col = c(rep("red", 4), rep("blue", 4), rep("green", 4), rep("yellow", 4)),
+          main="Distribución de los valores de intensidad de los Arrays")
```

Una vez más, E.C.3 y E.SP3.4 son con diferencia los que más variabilidad contienen. El siguiente paso por tanto es ver si esta variabilidad es fruto de un error técnico o si se pueden realizar comparaciones con la normalización de los datos.

Normalización de los datos

Para efectuar la normalización vamos a emplear el método más indicado para microarrays de Affymetrix, el RMA de Bioconductor que és el método estándar:

```
> eset_rma <- rma(rawData)
```

```
Background correcting
Normalizing
Calculating Expression
```

Control de calidad de los datos normalizados

Al efectuar el análisis de control de calidad de los dtos una vez normalizados, vemos que se verifica la disposición de "outlier" el Array E.C.3. Sin embargo, el array E.SP3.4 ya no aparece en la lista. Estos resultados pueden corroborarse con otro Análisis de Componentes Principales para los datos estandarizados;

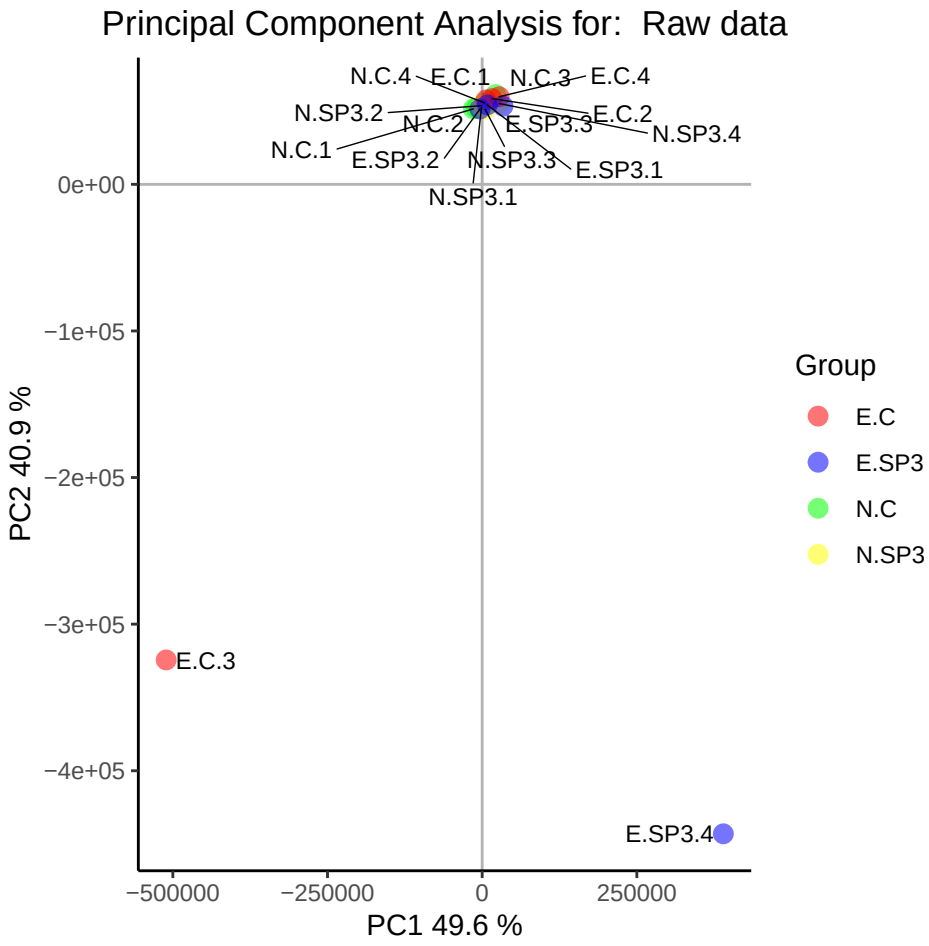


Figure 2: Análisis de Componentes Principales de rawData

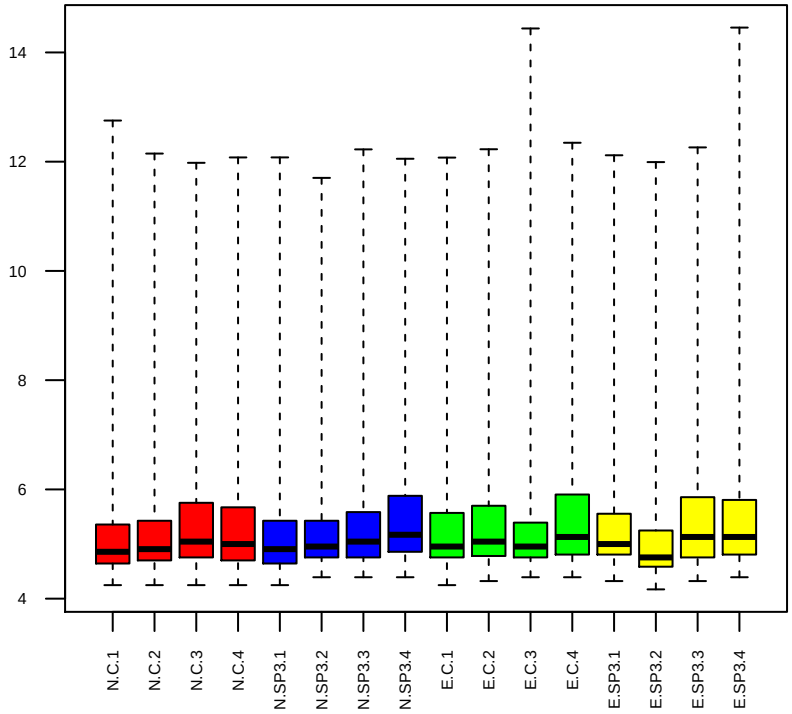


Figure 3: Boxplot de los Arrays con datos brutos

```
> #arrayQualityMetrics(eset_rma, outdir = file.path("./results", "QCDir.Norm"), force=TRUE)
```

```
> plotPCA3(exprs(eset_rma), labels = eset_rma@phenoData@data[["ShortName"]], factor = eset_rma@phenoData
+          title="Datos normalizados", scale = FALSE, size = 3,
+          colores = c("red", "blue", "green", "yellow"))
```

Con los datos normalizados vemos que el eje que explica el 22.9% de la variabilidad separa a los controles de los casos inoculados con SP3. Además podemos apreciar que el Array E.C.3 sigue estando bastante aislado del resto de sus pares. Por otra parte también podemos observar que el eje que explica un 18.2% de la variabilidad de los Arrays separa, aunque con una menor claridad, los casos "naive" (no expuestos previamente a la bacteria) de los experimentados (ratones inoculados con SP3 y recuperados previamente al experimento).

```
> boxplot(eset_rma, cex.axis=0.5, las=2,  which="all",
+          col = c(rep("red", 4), rep("blue", 4), rep("green", 4), rep("yellow", 4)),
+          main="Boxplot de intensidades de los Arrays: Datos normalizados")
```

En este análisis se pueden apreciar mucha más similaridad en las intensidades de todos los Arrays (a excepción una vez mas del Array E.C.3).

Filtraje no específico

A continuación vamos a efectuar el análisis del posible ruido de fondo que puedan tener los dieferentes Arrays del estudio. Vamos a emplear el análisis de componentes principales de variación;

	array	sampleNames	*1	*2	*3	Group	Experimental	Cell.type	ShortName
☐	1	N.C.1				N.C	Control	Naive	N.C.1
☐	2	N.C.2				N.C	Control	Naive	N.C.2
☐	3	N.C.3				N.C	Control	Naive	N.C.3
☐	4	N.C.4				N.C	Control	Naive	N.C.4
☐	5	N.SP3.1				N.SP3	SP3	Naive	N.SP3.1
☐	6	N.SP3.2				N.SP3	SP3	Naive	N.SP3.2
☐	7	N.SP3.3				N.SP3	SP3	Naive	N.SP3.3
☐	8	N.SP3.4				N.SP3	SP3	Naive	N.SP3.4
☐	9	E.C.1				E.C	Control	Experienced	E.C.1
☐	10	E.C.2				E.C	Control	Experienced	E.C.2
☑	11	E.C.3	x	x		E.C	Control	Experienced	E.C.3
☐	12	E.C.4				E.C	Control	Experienced	E.C.4
☐	13	E.SP3.1				E.SP3	SP3	Experienced	E.SP3.1
☑	14	E.SP3.2		x		E.SP3	SP3	Experienced	E.SP3.2
☐	15	E.SP3.3				E.SP3	SP3	Experienced	E.SP3.3
☐	16	E.SP3.4				E.SP3	SP3	Experienced	E.SP3.4

Figure 4: Tabla resumen de los datos del estudio normalizados

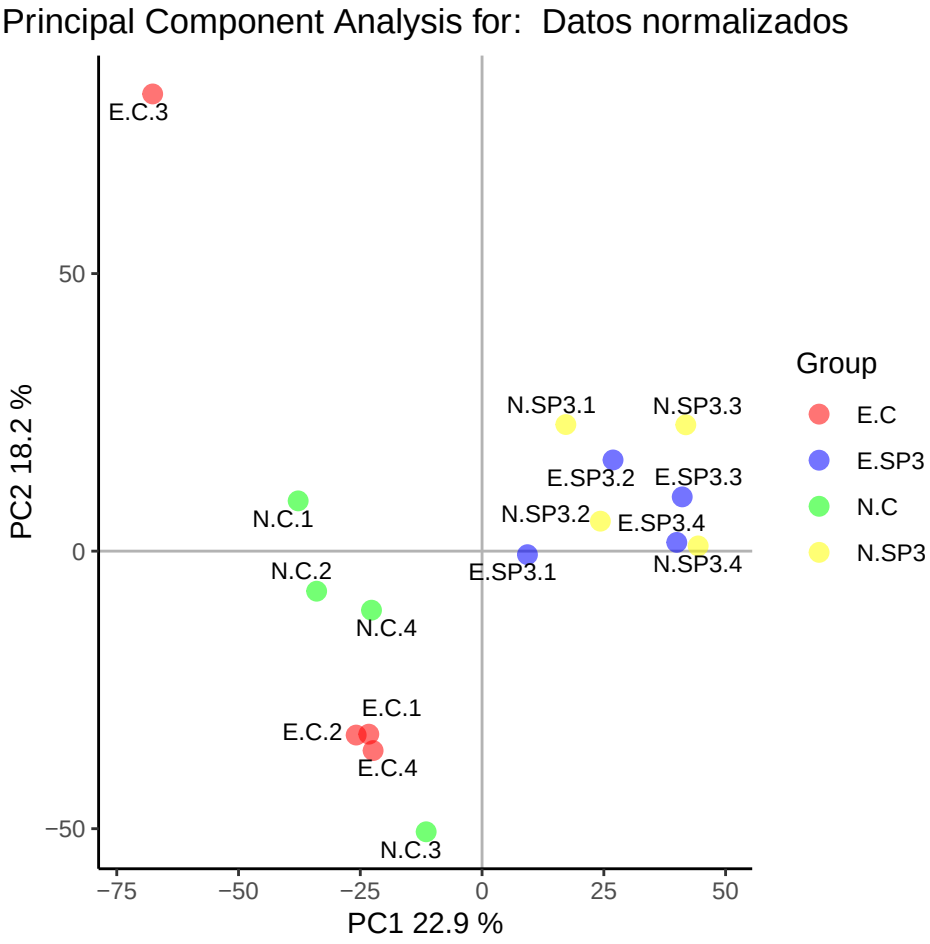


Figure 5: Análisis de Componentes Principales de los datos normalizados

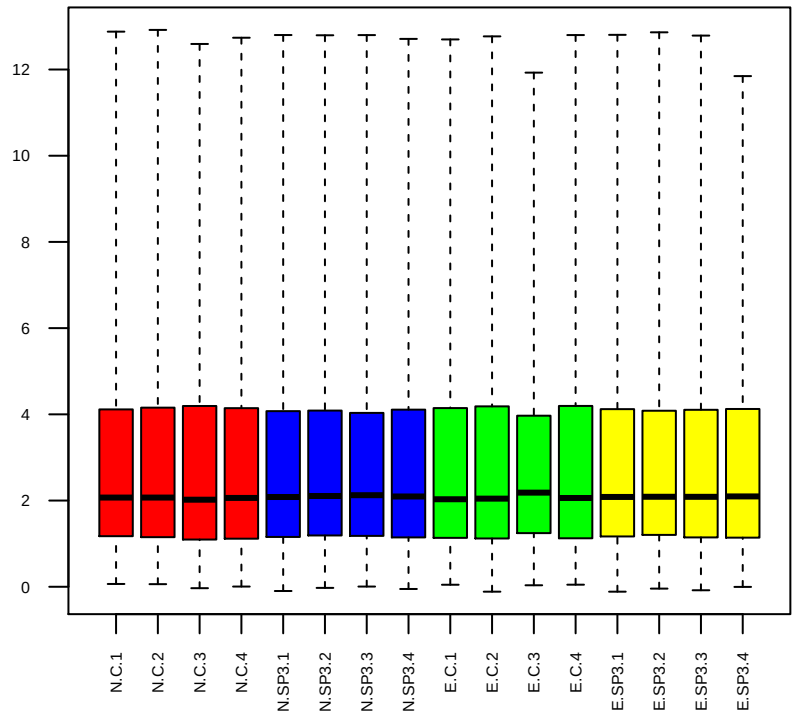
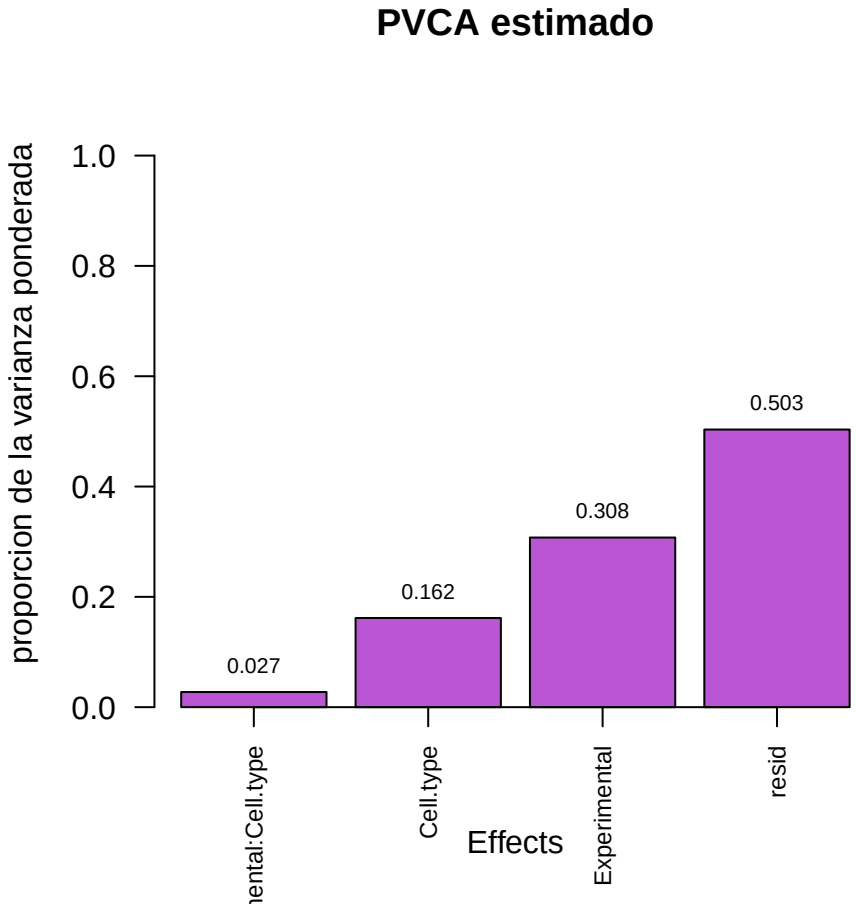


Figure 6: Boxplot con datos normalizados

```
> pData(eset_rma) <- my.targets@data
> pct_threshold <- 0.6
> batch.factors <- c("Experimental", "Cell.type")
> pvcaObj <- pvcaBatchAssess (eset_rma, batch.factors, pct_threshold)
```

```
> bp <- barplot(pvcaObj$dat, xlab = "Effects",
+   ylab = "proporcion de la varianza ponderada",
+   ylim= c(0,1.1),col = c("mediumorchid"), las=2,
+   main="PVCA estimado")
> axis(1, at = bp, labels = pvcaObj$label, cex.axis = 0.75, las=2)
> values = pvcaObj$dat
> new_values = round(values , 3)
> text(bp,pvcaObj$dat,labels = new_values, pos=3, cex = 0.7)
```



Como se puede observar, existe una mayor variación no biológica en este caso que es la residual. Esto hace suponer que deberían eliminarse al menos los Arrays que más ruido puedan presentar (por ejemplo el Array E.C.3). En segundo lugar hemos corroborado que es la variabilidad explicada por el grupo experimental (diferencias entre control y experienced). Y por último, un 16% de la variabilidad sería explicada por el factor de interés en este estudio que no es otro que la diferencia entre las células "naive" y las experimentadas.

Identificación de genes diferencialmente expresados

Viendo los resultados del apartado anterior, hay serias sospechas de que la variabilidad total de las muestras pueda enmascarar las diferencias entres los distintos grupos experimentales. Por este mitivo se procede a realizar una análisis de la variabilidad de todos los genes para ver que porcentaje de genes pueden mostrar una variabilidad distinta a la varibilidad genérica de las muestras:

```
> sds <- apply (exprs(eset_rma), 1, sd)
> sds0<- sort(sds)
> plot(1:length(sds0), sds0, main="Distribución de la variabilidad de los genes",
+      sub="Vertical lines represent 90% and 95% percentiles",
+      xlab="Gene index (from least to most variable)", ylab="Standard deviation")
> abline(v=length(sds)*c(0.9,0.95))
```

Así podemos apreciar que existe un gran número de genes que presentan una desviación estándar mayor a 1.0, aunque no llega a ser el 5% de los genes.

Filtrado de los genes menos variables

El siguiente paso que vamos a realizar es tratar de eliminar todos aquellos genes que puedan provocar una mayor distorsión a la hora de realizar los análisis comparativos;

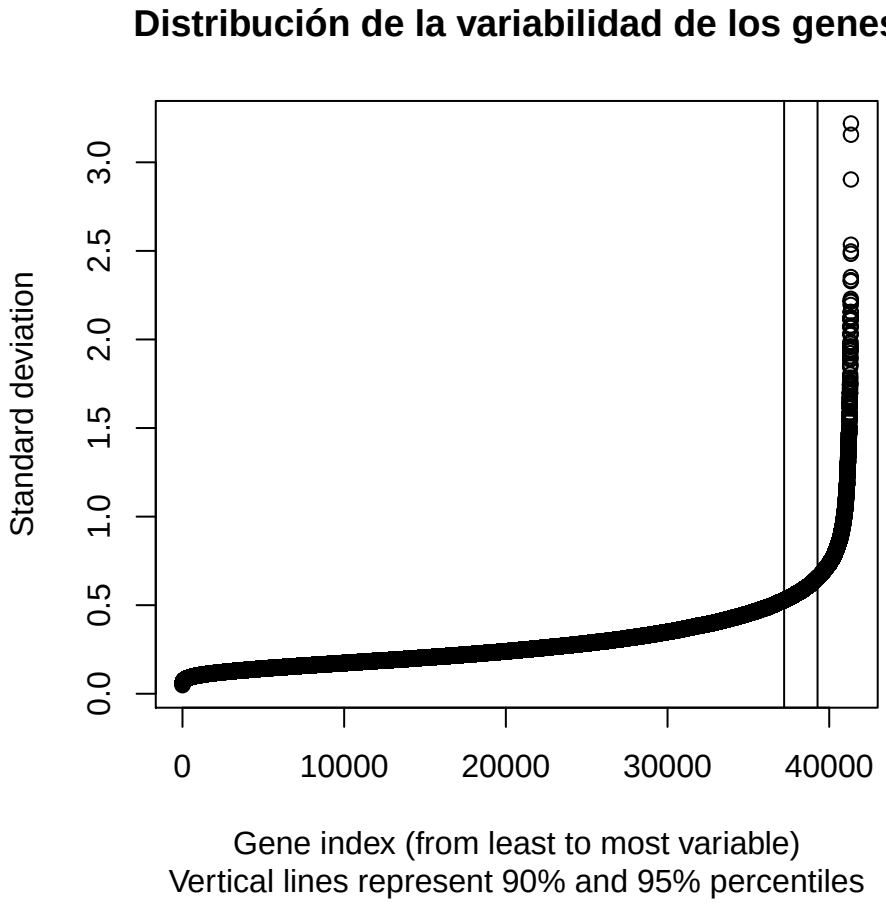


Figure 7: Valores de desviaciones estándar de todas las muestras para todos los genes

```
> annotation(eset_rma) <- "mogene21sttranscriptcluster.db"
> filtered <- nsFilter(eset_rma,
+                      require.entrez = TRUE, remove.dupEntrez = TRUE,
+                      var.filter=TRUE, var.func=IQR, var.cutoff=0.75,
+                      filterByQuantile=TRUE, feature.exclude = "^AFFX")
```

```
> print(filtered$filter.log)
```

```
$numDupsRemoved
[1] 671

$numLowVar
[1] 17973

$numRemoved.ENTREZID
[1] 16710
```

```
> eset_filtered <-filtered$eset
```

Con lo que con esta función se han filtrado los genes, eliminando un total de 16710 lo cual es bastante considerable y de suponer teniendo en cuenta los análisis previos. Acontinuación guardamos el archivo obtenido para posibles análisis posteriores;

```
> 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")
```

Matriz de diseño

A continuación vamos a construir el modelo In this study that "Group" variable is a combination of the two experimental conditions, "KO/Wild" and "RT/COLD" which are jointly represented as one factor with 4 levels.

```
> if (!exists("eset_filtered")) load (file="./results/normalized.Data.Rda")
```

```
> designMat<- model.matrix(~0+Group, pData(eset_filtered))
> colnames(designMat) <- c("E.C", "E.SP3", "N.C", "N.SP3")
> print(designMat)
```

```
              E.C E.SP3 N.C N.SP3
GSM3931536.CEL   0     0   1     0
GSM3931537.CEL   0     0   1     0
GSM3931538.CEL   0     0   1     0
GSM3931539.CEL   0     0   1     0
GSM3931540.CEL   0     0   0     1
GSM3931541.CEL   0     0   0     1
GSM3931542.CEL   0     0   0     1
GSM3931543.CEL   0     0   0     1
GSM3931544.CEL   1     0   0     0
GSM3931545.CEL   1     0   0     0
GSM3931546.CEL   1     0   0     0
GSM3931547.CEL   1     0   0     0
GSM3931548.CEL   0     1   0     0
GSM3931549.CEL   0     1   0     0
GSM3931550.CEL   0     1   0     0
GSM3931551.CEL   0     1   0     0
```

```
attr(,"assign")
[1] 1 1 1 1
attr(,"contrasts")
attr(,"contrasts")$Group
[1] "contr.treatment"
```

```
> cont.matrix <- makeContrasts (Infec.E = E.C-E.SP3,
+                               Infec.N = N.C-N.SP3,
+                               MEM = (E.C-E.SP3) - (N.C-N.SP3),
+                               levels=designMat)
> print(cont.matrix)
```

```
       Contrasts
Levels  Infec.E Infec.N MEM
  E.C         1       0   1
  E.SP3      -1       0  -1
  N.C         0       1  -1
  N.SP3       0      -1   1
```

En este estudio y dado el diseño del mismo, lo que interesa aquí es el efecto memoria (llamado aquí MEM) que establecen los macrófagos al experimentar previamente la infección del SP3. No obstante será interesante tambien examinar qué efecto tiene la infección de SP3 en la expresión de genes de los macrófagos de ambos grupos:

Identificación de genes diferencialmente expresados

Vamos a obtener el listado de genes que se encuentran expresados de manera diferente según los tres casos que hemos diseñado en el apartado anterior. Para ello hemos de comprobar las pruebas de significación para cada gen y cada comparación:

```
> fit<-lmFit(eset_filtered, designMat)
> fit.main<-contrasts.fit(fit, cont.matrix)
> fit.main<-eBayes(fit.main)
> class(fit.main)
```

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

Y con la función "Toptable" se podrán generar para cada contraste una lista de genes ordenados de mayor a menor diferencia de expresión:

Para los macrófacos experimentados;

```
> topTab_Infec.E <- topTable (fit.main, number=nrow(fit.main), coef="Infec.E", adjust="fdr")
> head(topTab_Infec.E)
```

```
            logFC  AveExpr         t      P.Value    adj.P.Val        B
17438987 -3.778553 6.270982 -26.41976 2.946531e-15 1.765267e-11 23.58938
17344309 -3.538061 9.426964 -17.26318 3.197329e-12 9.577600e-09 17.90463
17527016 -2.919567 5.690399 -14.45400 5.468506e-11 1.092061e-07 15.33612
17487457 -2.111985 5.969117 -13.82153 1.106635e-10 1.510013e-07 14.68127
17213192 -2.171798 6.812150 -13.67856 1.302712e-10 1.510013e-07 14.52887
17376153 -2.767861 4.017678 -13.54891 1.512281e-10 1.510013e-07 14.38923
```

Para los "naive";

```
> topTab_Infec.N <- topTable (fit.main, number=nrow(fit.main), coef="Infec.N", adjust="fdr")
> head(topTab_Infec.N)
```

```
            logFC  AveExpr         t      P.Value    adj.P.Val        B
17438987 -3.688876 6.270982 -25.79274 4.393042e-15 2.631871e-11 23.31176
17246091 -4.993739 4.321504 -14.87421 3.473426e-11 8.496398e-08 15.75903
17213192 -2.331613 6.812150 -14.68512 4.254581e-11 8.496398e-08 15.57222
17527016 -2.866409 5.690399 -14.19083 7.308629e-11 1.094650e-07 15.07134
17344309 -2.824384 9.426964 -13.78095 1.158905e-10 1.388599e-07 14.64171
17376153 -2.685823 4.017678 -13.14733 2.419115e-10 2.415486e-07 13.95072
```

Para el contraste de mayor interés, la diferencia entre los experimentados y "naive", o efecto memoria (MEM);

```
> topTab_MEM  <- topTable (fit.main, number=nrow(fit.main), coef="MEM", adjust="fdr")
> head(topTab_MEM)
```

```
            logFC  AveExpr         t      P.Value  adj.P.Val          B
17447013 -1.966180 2.703197 -6.432623 6.154179e-06 0.03496504  1.19491666
17467415  1.756991 2.401136  6.102050 1.167252e-05 0.03496504  0.90511732
17221627 -2.509283 3.825326 -5.265138 6.291006e-05 0.12563140  0.08863356
17352517 -1.563086 1.970251 -5.079013 9.256584e-05 0.13864049 -0.10935220
17366992 -1.283150 3.172075 -4.628628 2.392466e-04 0.25879676 -0.61258183
17211369 -2.250983 3.568376 -4.591066 2.591855e-04 0.25879676 -0.65604486
```

Anotación de los resultados obtenidos

En este apartado vamos a correlacionar las etiquetas de los genes o ID. establecidas por Affimetrix con los *gene symbol* establecidos para la descripción de cada uno de los genes:

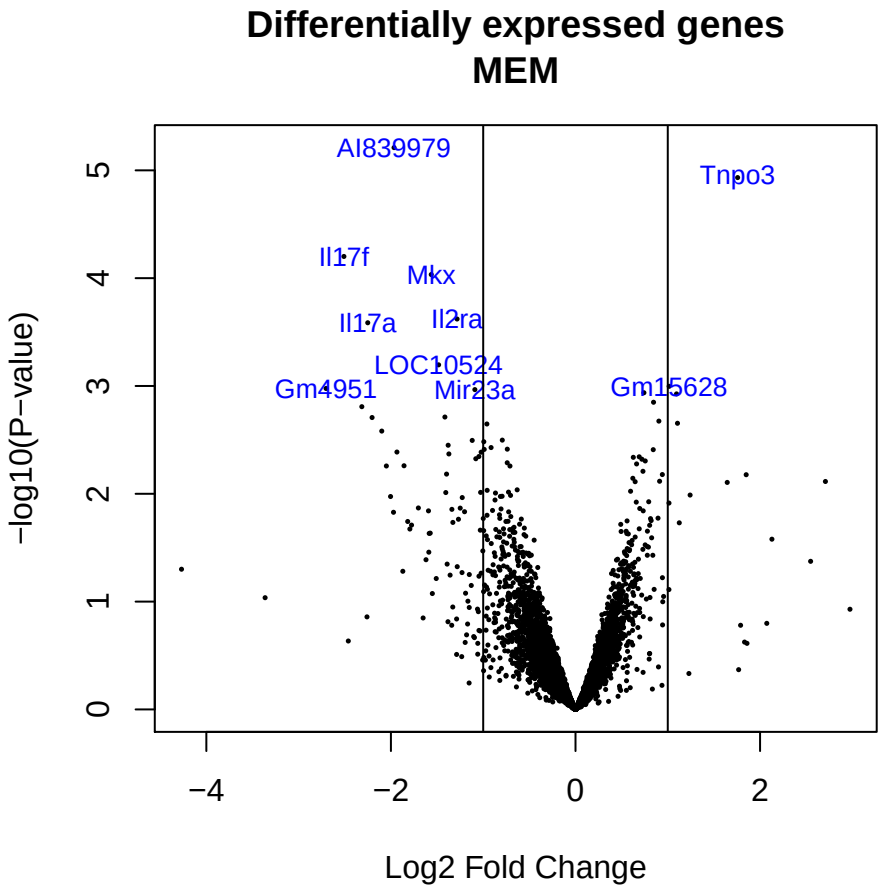
```
> annotatedTopTable <- function(topTab, anotPackage)
+ {
+   topTab <- cbind(PROBEID=rownames(topTab), topTab)
+   myProbes <- rownames(topTab)
+   thePackage <- eval(parse(text = anotPackage))
+   geneAnots <- select(thePackage, myProbes, c("SYMBOL", "ENTREZID", "GENENAME"))
+   annotatedTopTab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

```
> topAnnotated_Infec.E <- annotatedTopTable(topTab_Infec.E,
+ anotPackage="mogene21sttranscriptcluster.db")
> topAnnotated_Infec.N <- annotatedTopTable(topTab_Infec.N,
+ anotPackage="mogene21sttranscriptcluster.db")
> topAnnotated_MEM <- annotatedTopTable(topTab_MEM,
+ anotPackage="mogene21sttranscriptcluster.db")
> write.csv(topAnnotated_Infec.E, file="./results/topAnnotated_Infec.E.csv")
> write.csv(topAnnotated_Infec.N, file="./results/topAnnotated_Infec.N.csv")
> write.csv(topAnnotated_MEM, file="./results/topAnnotated_MEM.csv")
> show(head(topAnnotated_MEM[1:5,1:4]))
```

```
   PROBEID SYMBOL ENTREZID                                        GENENAME
1 17210912 Rb1cc1    12421                     RB1-inducible coiled-coil 1
2 17211000   Rrs1    59014                   ribosome biogenesis regulator 1
3 17211004 Adhfe1    76187        alcohol dehydrogenase, iron containing, 1
4 17211043   Sgk3   170755          serum/glucocorticoid regulated kinase 3
5 17211090  Cspp1   211660 centrosome and spindle pole associated protein 1
```

Y visualizamos también los datos mediante un volcanoPlot de las diferencias de memoria entre los macrófagos infectados:

```
> library(mogene21sttranscriptcluster.db)
> geneSymbols <- select(mogene21sttranscriptcluster.db, rownames(fit.main), c("SYMBOL"))
> SYMBOLS<- geneSymbols$SYMBOL
> volcanoplot(fit.main, coef=3, highlight=10, names=SYMBOLS,
+             main=paste("Differentially expressed genes", colnames(cont.matrix)[3], sep="\n"))
>   abline(v=c(-1,1))
```



Comparación entre distintas comparaciones

Aunque el principal motivo de este estudio es la comparación MEM, puede resultar de interés qué sucede en el resto de comparaciones con los controles. El siguiente paso es ver si hay genes están seleccionados entre las combinaciones de las comparaciones efectuadas en este estudio:

```
> library(limma)
> res<-decideTests(fit.main, method="separate", adjust.method="fdr", p.value=0.1, lfc=1)
```

Con el valor +1 para los sobreexpresados y el valor -1 para los infraexpresados el punto de corte para el análisis se define como "FDR < 0.1" y "logFC > 1" (+1 para valores de t-test > 0, FDR < punto de corte y -1 para valores de t-test < 0, FDR < punto de corte) tomando valores 0 para no diferencias significativas.

```
> sum.res.rows<-apply(abs(res),1,sum)
> res.selected<-res[sum.res.rows!=0,]
> print(summary(res))
```

```
       Infec.E Infec.N  MEM
Down       294     253    1
NotSig    5532    5471 5989
```

```
Up          165     267     1
```

Podemos observar que existe un gen sobreexpresado y otro infraexpresado que comparten el efecto memoria con los macrófagos experimentados, esto es muy informativo y llama bastante la atención. Otra manera más simplificada y visual de representar esta información sería mediante un Diagrama de Venn:

```
> vennDiagram (res.selected[,1:3], cex=0.9)
> title("Genes en común entre los tres grupos definidos\n Genes seleccionados mediante FDR < 0.1 y logF
```

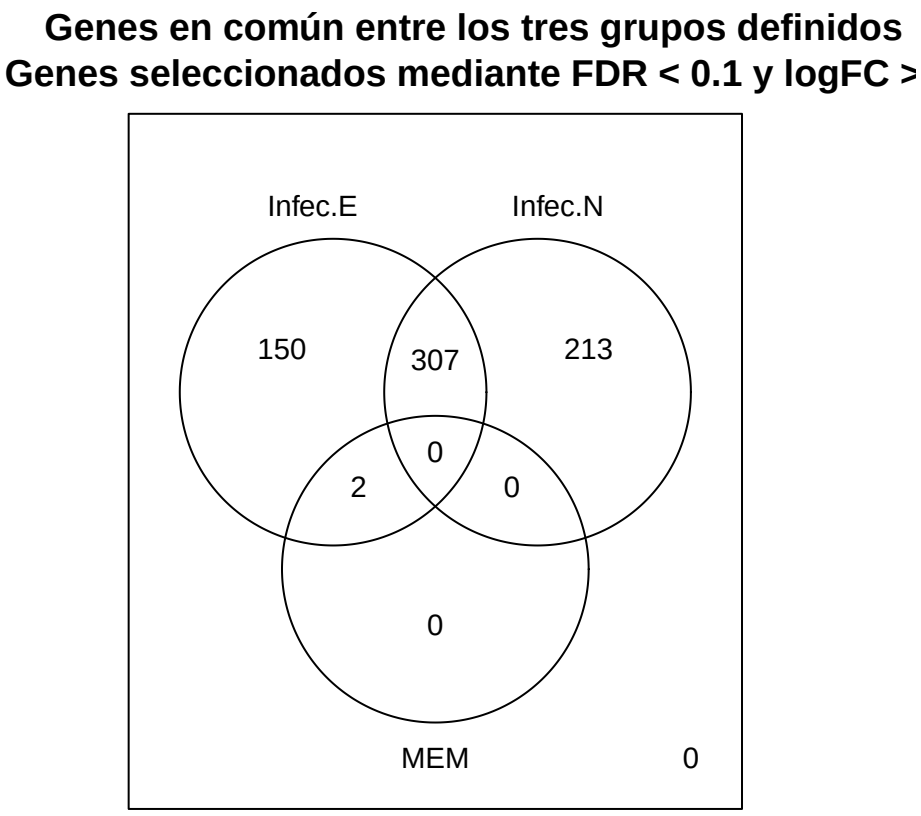


Figure 8: Venn diagram showing the genes in common between the three comparisons performed

En este diagrama se observa el número de genes expresados diferentemente de manera significativa, pero no se ven si están infra o sobreexpresados. Para ver la información con más detalle podemos realizar el Heatmap:

```
> probesInHeatmap <- rownames(res.selected)
> HMdata <- exprs(eset_filtered)[rownames(exprs(eset_filtered)) %in% probesInHeatmap,]
>
> geneSymbols <- select(mogene21sttranscriptcluster.db, rownames(HMdata), c("SYMBOL"))
> SYMBOLS<- geneSymbols$SYMBOL
> rownames(HMdata) <- SYMBOLS
> write.csv(HMdata, file = file.path("./results/data4Heatmap.csv"))
```

```
> my_palette <- colorRampPalette(c("blue", "red"))(n = 299)
> heatmap.2(HMdata,
+           Rowv = TRUE,
+           Colv = TRUE,
+           dendrogram = "both",
+           main = "Genes expresados diferente \n FDR < 0,1, logFC >=1",
+           scale = "row",
```

```
+           col = my_palette,
+           sepcolor = "white",
+           sepwidth = c(0.05,0.05),
+           cexRow = 0.5,
+           cexCol = 0.9,
+           key = TRUE,
+           keysize = 1.5,
+           density.info = "histogram",
+           ColSideColors = c(rep("red",4),rep("blue",4), rep("green",4), rep("yellow",4)),
+           tracecol = NULL,
+           srtCol = 30)
```

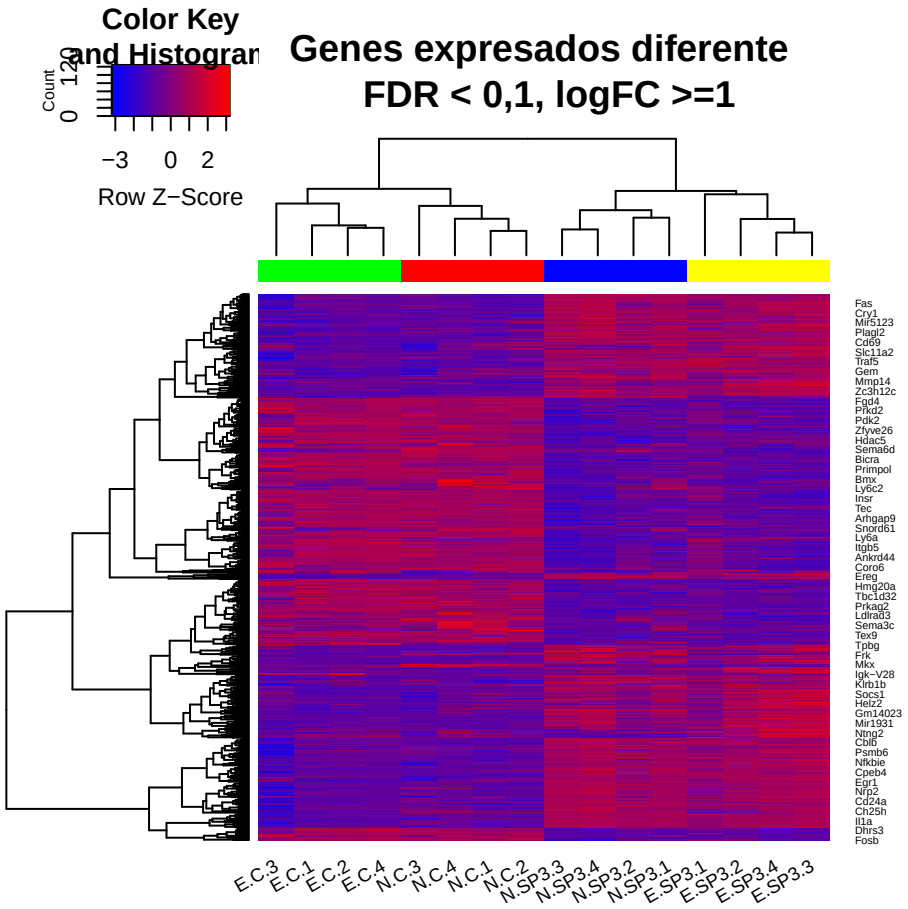


Figure 9: Heatmap de similaridad para expresión de genes (filas) y muestras (columnas)

Análisis de significación biológica

En este paso trataremos de ver en qué vía metabolica tienen en común los gens que están diferenciados significativamente entre los grupos analizados. Lo vamos a realizar con el paquete `ReactomePA` de Bioconductor con un FDR < 0.15 (menos restrictivo) para tratar de asegurar el listado de genes suficientes para poder llegar a coincidencias en vías metabólicas.

```
> listOfTables <- list(Infec.E = topTab_Infec.E,
+                      Infec.N  = topTab_Infec.N,
+                      MEM = topTab_MEM)
```

```
> listOfSelected <- list()
> for (i in 1:length(listOfTables)){
+   # select the toptable
+   topTab <- listOfTables[[i]]
+   # select the genes to be included in the analysis
+   whichGenes<-topTab["adj.P.Val"]<0.15
+   selectedIDs <- rownames(topTab)[whichGenes]
+   # convert the ID to Entrez
+   EntrezIDs<- select(mogene21sttranscriptcluster.db, selectedIDs, c("ENTREZID"))
+   EntrezIDs <- EntrezIDs$ENTREZID
+   listOfSelected[[i]] <- EntrezIDs
+   names(listOfSelected)[i] <- names(listOfTables)[i]
+ }
> sapply(listOfSelected, length)
```

```
Infec.E Infec.N     MEM
   2632    3023       4
```

```
> mapped_genes2GO <- mappedkeys(org.Mm.egGO)
> mapped_genes2KEGG <- mappedkeys(org.Mm.egPATH)
> mapped_genes <- union(mapped_genes2GO , mapped_genes2KEGG)
```

```
> listOfData <- listOfSelected[1:2]
> comparisonsNames <- names(listOfData)
> universe <- mapped_genes
>
> for (i in 1:length(listOfData)){
+   genesIn <- listOfData[[i]]
+   comparison <- comparisonsNames[i]
+   enrich.result <- enrichPathway(gene = genesIn,
+                                  pvalueCutoff = 0.05,
+                                  readable = T,
+                                  pAdjustMethod = "BH",
+                                  organism = "mouse",
+                                  universe = universe)
+
+   cat("####################################")
+   cat("\nComparison: ", comparison,"\n")
+   print(head(enrich.result))
+
+   if (length(rownames(enrich.result@result)) != 0) {
+   write.csv(as.data.frame(enrich.result),
+             file =paste0("./results/","ReactomePA.Results.",comparison,".csv"),
+             row.names = FALSE)
+
+   pdf(file=paste0("./results/","ReactomePABarplot.",comparison,".pdf"))
+     print(barplot(enrich.result, showCategory = 15, font.size = 4,
+             title = paste0("Reactome Pathway Analysis for ", comparison,". Barplot")))
+   dev.off()
+
+   pdf(file = paste0("./results/","ReactomePAcnetplot.",comparison,".pdf"))
+     print(cnetplot(enrich.result, categorySize = "geneNum", schowCategory = 15,
+          vertex.label.cex = 0.75))
+   dev.off()
```

```
+   }
+ }

###################################
Comparison:  Infec.E
                         ID                                     Description
R-MMU-1483255 R-MMU-1483255                                  PI Metabolism
R-MMU-168898   R-MMU-168898                      Toll-like Receptor Cascades
R-MMU-1660516 R-MMU-1660516 Synthesis of PIPs at the early endosome membrane
R-MMU-449147   R-MMU-449147                        Signaling by Interleukins
R-MMU-168138   R-MMU-168138             Toll Like Receptor 9 (TLR9) Cascade
R-MMU-5617833 R-MMU-5617833                                Cilium Assembly
              GeneRatio  BgRatio       pvalue     p.adjust       qvalue
R-MMU-1483255   27/1226  74/8772 1.003952e-06 0.0009828691 0.0008655124
R-MMU-168898    39/1226 133/8772 2.939510e-06 0.0013779552 0.0012134244
R-MMU-1660516   10/1226  15/8772 4.222539e-06 0.0013779552 0.0012134244
R-MMU-449147    63/1226 262/8772 6.640942e-06 0.0016253706 0.0014312978
R-MMU-168138    26/1226  85/8772 5.871189e-05 0.0107598123 0.0094750672
R-MMU-5617833   48/1226 198/8772 6.594369e-05 0.0107598123 0.0094750672

R-MMU-1483255
R-MMU-168898
R-MMU-1660516
R-MMU-449147  Il1a/Il23a/Nfkb2/Nfkb1/Nfkbia/Il6/Nfkbib/Socs3/Il2ra/Mapkapk2/Syk/Rela/Irak3/Il20rb/Stat5
R-MMU-168138
R-MMU-5617833                                                                                  Ift140/Sd
              Count
R-MMU-1483255    27
R-MMU-168898     39
R-MMU-1660516    10
R-MMU-449147     63
R-MMU-168138     26
R-MMU-5617833    48
###################################
Comparison:  Infec.N
                         ID                                     Description
R-MMU-69278     R-MMU-69278                          Cell Cycle, Mitotic
R-MMU-1483255 R-MMU-1483255                                  PI Metabolism
R-MMU-1660516 R-MMU-1660516 Synthesis of PIPs at the early endosome membrane
R-MMU-174417   R-MMU-174417     Telomere C-strand (Lagging Strand) Synthesis
R-MMU-69190     R-MMU-69190                        DNA strand elongation
R-MMU-69242     R-MMU-69242                                    S Phase
              GeneRatio  BgRatio       pvalue     p.adjust       qvalue
R-MMU-69278    122/1394 499/8772 2.311884e-07 0.0002323444 0.0002049059
R-MMU-1483255   29/1394  74/8772 1.081416e-06 0.0005434116 0.0004792381
R-MMU-1660516   10/1394  15/8772 1.385912e-05 0.0046428052 0.0040945190
R-MMU-174417    12/1394  22/8772 3.365045e-05 0.0067637400 0.0059649847
R-MMU-69190     12/1394  22/8772 3.365045e-05 0.0067637400 0.0059649847
R-MMU-69242     41/1394 142/8772 6.011900e-05 0.0100699319 0.0088807360

R-MMU-69278   Cdkn1b/Dna2/Cdc25b/Pds5b/Numa1/Optn/Vrk1/Sdccag8/Mad1l1/Rbl1/Pole2/Ccnd3/Cep192/Ncapd2/Rf
R-MMU-1483255
R-MMU-1660516
```

```
R-MMU-174417
R-MMU-69190
R-MMU-69242
              Count
R-MMU-69278     122
R-MMU-1483255    29
R-MMU-1660516    10
R-MMU-174417     12
R-MMU-69190      12
R-MMU-69242      41
```

```
>   cnetplot(enrich.result, categorySize = "geneNum", schowCategory = 15,
+          vertex.label.cex = 0.75)
```

Figure 10: Red obtenida del análisis Reactome enrichment de la lista obtenida de las comparaciones de Naive Experienced y MEM

Table 1: Primeras filas y columnas de los resultados de Reactome de la comparacion de Infec.E.csv

	Description	GeneRatio	BgRatio	pvalue
R-MMU-1483255	PI Metabolism	27/1226	74/8772	1.00395205521062e-06
R-MMU-168898	Toll-like Receptor Cascades	39/1226	133/8772	2.93951028706845e-06

	Description	GeneRatio	BgRatio	pvalue
R-MMU-1660516	Synthesis of PIPs at the early endosome membrane	10/1226	15/8772	4.22253895303042e-06
R-MMU-449147	Signaling by Interleukins	63/1226	262/8772	6.64094204174427e-06

Es interesante que la vía metabólica más relevante en los cambios genéticos sea la vía del metabolismo de los fosfolípidos de membrana seguida de la de cascadas de receptores Toll-like y de la de síntesis de fosfolípidos de la membrana de formación de endosomas, así como la de señalización de Interleukinas.

It is useful to create a file with the type, name and description of all the files generated along the analysis. Table @ref(tab:listOfFiles) shows the list of files generated in the current case study.

Table 2: Lista de archivos generados en el análisis

List_of_Files
data4Heatmap.csv
normalized.Data.csv
normalized.Data.Rda
normalized.Filtered.Data.csv
QCDir.Norm
ReactomePA.Results.Infec.E.csv
ReactomePA.Results.Infec.N.csv
ReactomePABarplot.Infec.E.pdf
ReactomePABarplot.Infec.N.pdf
ReactomePAcnetplot.Infec.E.pdf
ReactomePAcnetplot.Infec.N.pdf
topAnnotated_Infec.E.csv
topAnnotated_Infec.N.csv
topAnnotated_MEM.csv