Análisis de datos de transcriptómica

Alicia Peñaranda y Sergio Muñoz

14/6/2021

Instalation of the required packages

```
BiocManager::install(c('affy','limma','genefilter'))
BiocManager::install("hgu133plus2.db")
BiocManager::install("GSEABase")
BiocManager::install("fgsea")
BiocManager::install("ggplot")
```

First of all, we set the working directory

```
setwd("C:/Users/Alicia/OneDrive/Escritorio/Segundo cuatrimestre/transcriptómica/TREEP_ggomez/Trabajo1/GS
```

Quality control of the raw data: preprocessing and normalization

1. Load libraries

```
library("affy")
library("limma")
library("genefilter")
```

2. Import targets.txt file

This object contains the name of the different raw files that we are going to process. Six of the Raw.cel files contain the results after the treatment with DMSO (control) of both All and K1 cells. The other six Raw.cel files contain the results after the treatment with SAHM1 of All and K1 cells.

```
targets <- readTargets('targets.txt', row.names = 'FileName')
targets
```

```
##                     FileName       Classes
## GSM455115.CEL GSM455115.CEL  KOPT_K1_DMSO
## GSM455116.CEL GSM455116.CEL  KOPT_K1_DMSO
## GSM455117.CEL GSM455117.CEL  KOPT_K1_DMSO
## GSM455118.CEL GSM455118.CEL  HPB_ALL_DMSO
## GSM455119.CEL GSM455119.CEL  HPB_ALL_DMSO
## GSM455120.CEL GSM455120.CEL  HPB_ALL_DMSO
## GSM455121.CEL GSM455121.CEL KOPT_K1_SAHM1
```

```
## GSM455122.CEL GSM455122.CEL KOPT_K1_SAHM1
## GSM455123.CEL GSM455123.CEL KOPT_K1_SAHM1
## GSM455124.CEL GSM455124.CEL HPB_ALL_SAHM1
## GSM455125.CEL GSM455125.CEL HPB_ALL_SAHM1
## GSM455126.CEL GSM455126.CEL HPB_ALL_SAHM1
```

3. Import .CEL files

We create an AffyBatch object which we are going to work with. It contains the raw data and, as we can see in the output, the annotation is made using the hgu133plus2 database.

```
data <- ReadAffy(filenames = targets$FileName)
data
```

```
## AffyBatch object
## size of arrays=1164x1164 features (21 kb)
## cdf=HG-U133_Plus_2 (54675 affyids)
## number of samples=12
## number of genes=54675
## annotation=hgu133plus2
## notes=
```

4. Normalize with RMA

We've processed and normalized the Raw.cel applying the robust multiarray averaging (RMA) algorithm in order to background-correct, normalize and summarize background correction by creating an object of the class ExprSet.The result is a ExprSet object in which the intensities are provided in log scale.

```
eset <- expresso(data,
                 bg.correct = TRUE,
                 bgcorrect.method="rma",
                 normalize = TRUE,
                 normalize.method="quantiles",
                 pmcorrect.method="pmonly",
                 summary.method="medianpolish",
                 verbose = TRUE,
)
```

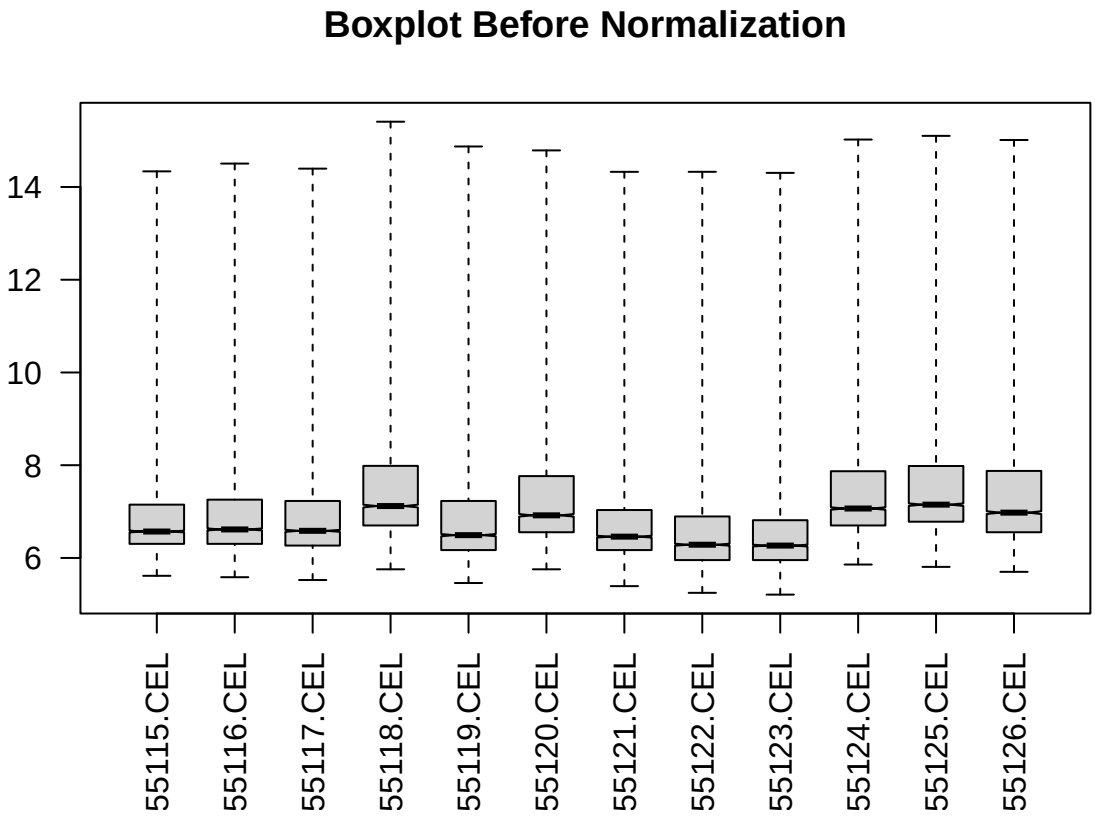
```
## background correction: rma
## normalization: quantiles
## PM/MM correction : pmonly
## expression values: medianpolish
## background correcting...done.
## normalizing...done.
## 54675 ids to be processed
## |                    |
## |####################|
```

It is a between array normalization in which we are using a quantiles method. With this method we ensure that the intensities have the same empirical distribution across arrays and across channels. This normalization process is different from the followed by the article as they performed it around the mean of the DMSO-treated samples on each pair gene basis.

5. Generate BOXPLOTS before and after normalization

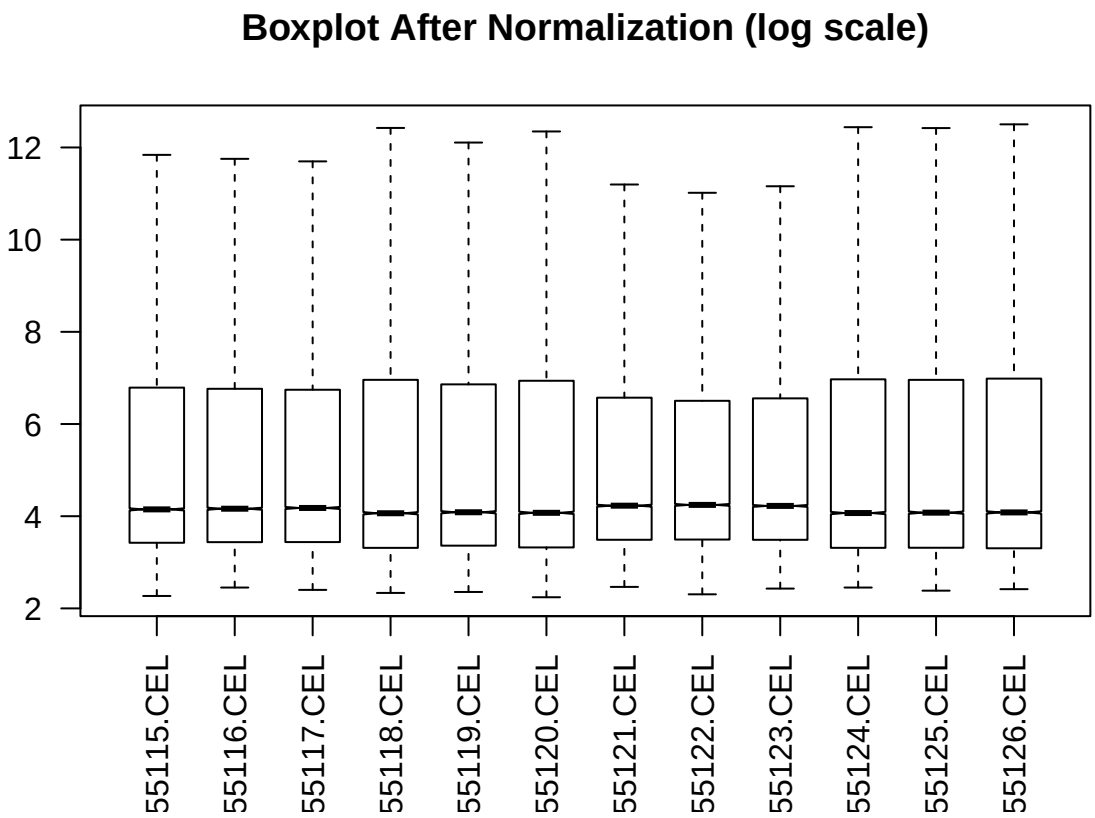
Boxplot for raw data:

```
boxplot(data,
        main="Boxplot Before Normalization",
        col = "lightgrey",
        outline=FALSE,
        boxwex=0.7,
        notch=T,
        las=2)
```



Boxplot for normalized data:

```
exprseset <- as.data.frame(exprs(eset))
boxplot(data.frame(exprseset),
        main="Boxplot After Normalization (log scale)",
        col = "white",
        outline=FALSE,
        boxwex=0.7,
        notch=T,
        las=2)
```



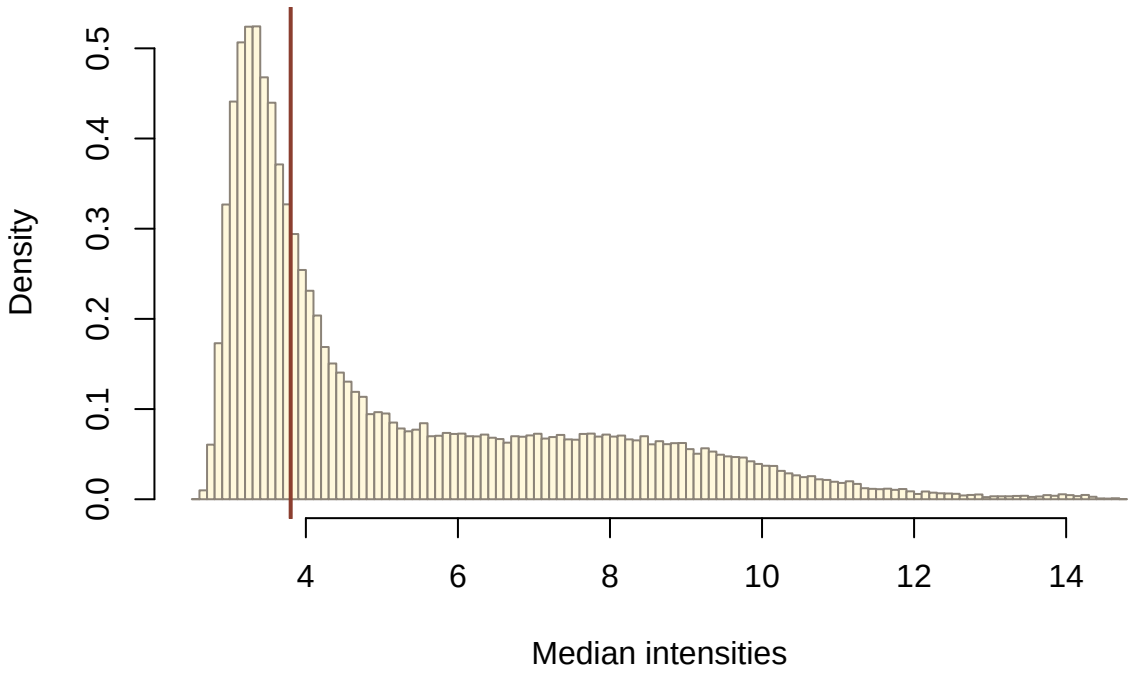
6. Data filtering using IQR

We filter the features exhibiting little variation, or a consistently low signal across samples. For that we use the IQR statistic function with a cutoff value of 0.5.

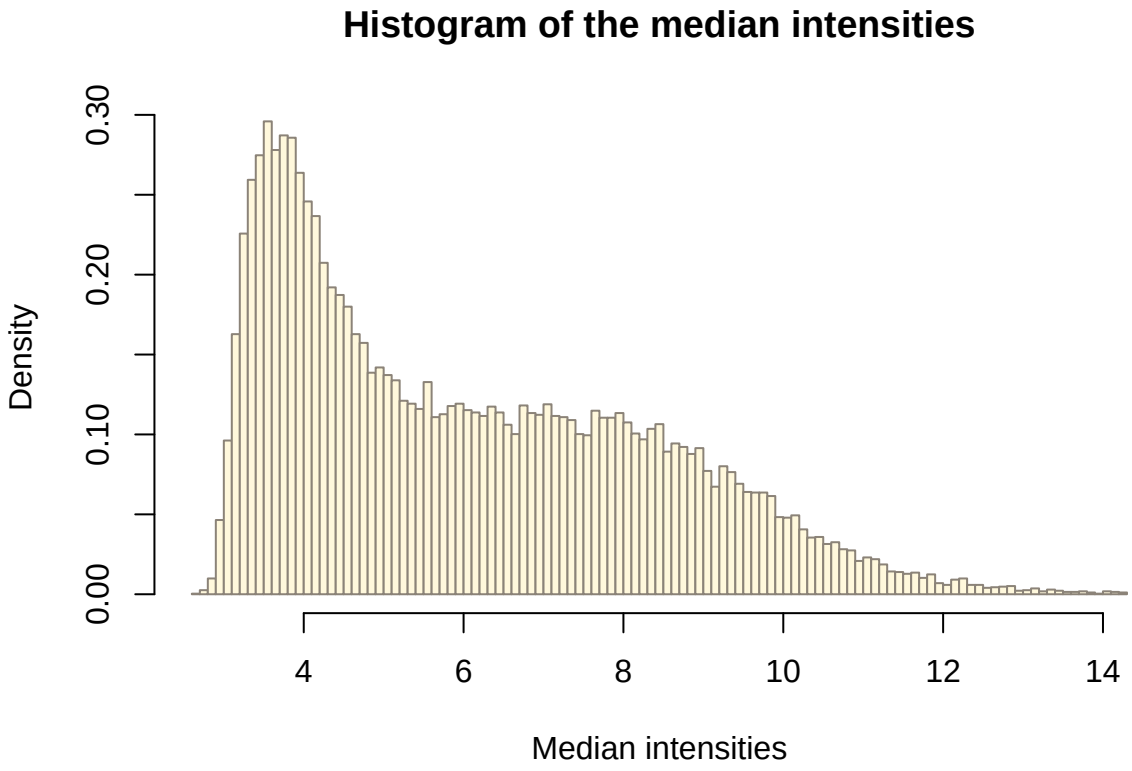
```
esetIQR <- varFilter(eset, var.func=IQR, var.cutoff=0.5, filterByQuantile=TRUE)
```

```
medians <- rowMedians(Biobase::exprs(eset))
hist_res <- hist(medians, 100, col = "cornsilk1", freq = FALSE,
                 main = "Histogram of the median intensities",
                 border = "antiquewhite4", xlab = "Median intensities")

abline(v = 3.8, col = "coral4", lwd = 2)
```




```
medians_IQR <- rowMedians(Biobase::exprs(esetIQR))
hist_res <- hist(medians_IQR, 100, col = "cornsilk1", freq = FALSE,
                 main = "Histogram of the median intensities",
                 border = "antiquewhite4", xlab = "Median intensities")
```



In the first histogram we see an enrichment of low medians of the left hand side. These are the genes we've filtered as we can see in the second histogram.

Differential expression analysis

1. Design matrix

We want to find differential expression between control and SAHM1 cells. First we create a design matrix.

```
KOPT_K1_DMSO <- c(1,1,1,0,0,0,0,0,0,0,0,0)
HPB_ALL_DMSO <- c(0,0,0,1,1,1,0,0,0,0,0,0)
KOPT_K1_SAHM1 <- c(0,0,0,0,0,0,1,1,1,0,0,0)
HPB_ALL_SAHM1 <- c(0,0,0,0,0,0,0,0,0,1,1,1)

design <-cbind(KOPT_K1_DMSO, HPB_ALL_DMSO, KOPT_K1_SAHM1, HPB_ALL_SAHM1)

rownames(design)<-targets$FileName
design
```

```
##               KOPT_K1_DMSO HPB_ALL_DMSO KOPT_K1_SAHM1 HPB_ALL_SAHM1
## GSM455115.CEL            1            0             0             0
## GSM455116.CEL            1            0             0             0
## GSM455117.CEL            1            0             0             0
## GSM455118.CEL            0            1             0             0
## GSM455119.CEL            0            1             0             0
```

```
## GSM455120.CEL            0            1            0            0
## GSM455121.CEL            0            0            1            0
## GSM455122.CEL            0            0            1            0
## GSM455123.CEL            0            0            1            0
## GSM455124.CEL            0            0            0            1
## GSM455125.CEL            0            0            0            1
## GSM455126.CEL            0            0            0            1
```

The rows are the raw.cel data and the columns are the variables we include in the linear model: type of cell and treatment. The design matrix entries are 1 or 0, depending on whether the variable is active or not.

2. Contrasts matrix

We use de design matrix to build the contrast matrix. That matrix will be used to obtain the differentially expressed genes.

```
cont.matrix<-makeContrasts(SAHM1vsDMSO=(KOPT_K1_SAHM1+HPB_ALL_SAHM1)-(KOPT_K1_DMSO+HPB_ALL_DMSO),
                           levels=design)

cont.matrix
```

```
##                Contrasts
## Levels          SAHM1vsDMSO
##   KOPT_K1_DMSO           -1
##   HPB_ALL_DMSO           -1
##   KOPT_K1_SAHM1           1
##   HPB_ALL_SAHM1           1
```

3. Obtaining differentialy expressed genes (DEGs)

- Linear model and eBayes

We fit a linear model to our data and apply the contrasts.fit function to it in order to find genes with significant differential expression between control and SAHM1 cells.

```
fit<-lmFit(esetIQR,design)  ##getting DEGs from IQR
fit2<-contrasts.fit(fit, cont.matrix)
fit2<-eBayes(fit2)
```

We applied the empirical Bayes variance moderation method to the model to improve the variance estimate.

Table with DEGs results:

```
toptableIQR<-topTable(fit2, number=dim(exprs(esetIQR))[1], adjust.method="BH")

head(toptableIQR)
```

```
##                 logFC  AveExpr         t      P.Value    adj.P.Val        B
## 201464_x_at  3.020346 6.934468  19.02309 1.628492e-10 4.451809e-06 13.80382
## 225520_at   -1.914647 9.538264 -17.15075 5.599354e-10 5.335427e-06 12.83218
## 209933_s_at -1.877392 8.547968 -16.97754 6.316459e-10 5.335427e-06 12.73465
## 227347_x_at -2.478369 6.970592 -16.56148 8.476674e-10 5.335427e-06 12.49462
## 201466_s_at  3.063284 4.708353  16.25781 1.055233e-09 5.335427e-06 12.31415
## 214079_at   -2.901964 4.173154 -16.11524 1.171034e-09 5.335427e-06 12.22783
```

4. Save results

```
save(toptableIQR,file="MyResultsNOTCH.RData")
```

Annotating gene lists using Bioconductor

1. Load annotation library

```
library("hgu133plus2.db")
library(GSEABase)
library(fgsea)
```

2. Retrieve Gene symbol

Annotate the summarized data with Gene Symbols. First we get the DEGs with an adjusted p.value <= 0.05.

```
load("MyResultsNOTCH.RData")
            #Array Affymetrix Human 430 v2
ID.fdr.005.table<-subset(toptableIQR, toptableIQR$adj.P.Val<=0.05)
head(ID.fdr.005.table)
```

```
##                 logFC  AveExpr         t      P.Value    adj.P.Val        B
## 201464_x_at  3.020346 6.934468  19.02309 1.628492e-10 4.451809e-06 13.80382
## 225520_at   -1.914647 9.538264 -17.15075 5.599354e-10 5.335427e-06 12.83218
## 209933_s_at -1.877392 8.547968 -16.97754 6.316459e-10 5.335427e-06 12.73465
## 227347_x_at -2.478369 6.970592 -16.56148 8.476674e-10 5.335427e-06 12.49462
## 201466_s_at  3.063284 4.708353  16.25781 1.055233e-09 5.335427e-06 12.31415
## 214079_at   -2.901964 4.173154 -16.11524 1.171034e-09 5.335427e-06 12.22783
```

Get Gene Symbols from DEGs (FDR<0.05) obtained.

```
probenames.fdr.005<-as.character(rownames(ID.fdr.005.table))
list.GeneSymbol.fdr.005<-mget(probenames.fdr.005, hgu133plus2SYMBOL, ifnotfound=NA)
char.GeneSymbol.fdr.005<- as.character(list.GeneSymbol.fdr.005)
toptable.annotated.symbol <-cbind(ID.fdr.005.table, char.GeneSymbol.fdr.005)
head(toptable.annotated.symbol)
```

```
##                 logFC  AveExpr         t      P.Value    adj.P.Val        B
## 201464_x_at  3.020346 6.934468  19.02309 1.628492e-10 4.451809e-06 13.80382
## 225520_at   -1.914647 9.538264 -17.15075 5.599354e-10 5.335427e-06 12.83218
## 209933_s_at -1.877392 8.547968 -16.97754 6.316459e-10 5.335427e-06 12.73465
## 227347_x_at -2.478369 6.970592 -16.56148 8.476674e-10 5.335427e-06 12.49462
## 201466_s_at  3.063284 4.708353  16.25781 1.055233e-09 5.335427e-06 12.31415
## 214079_at   -2.901964 4.173154 -16.11524 1.171034e-09 5.335427e-06 12.22783
##             char.GeneSymbol.fdr.005
## 201464_x_at                     JUN
## 225520_at                   MTHFD1L
## 209933_s_at                  CD300A
## 227347_x_at                    HES4
## 201466_s_at                     JUN
## 214079_at                    DHRS2
```

Here we show the top 50 downregulated genes (P<0.001). There are some of the genes of the paper in our top 50, as DXT1, HES1 or HES4. The gene MYC is downregulated but not in the top 50.

```
table.ordered = toptable.annotated.symbol[order(toptable.annotated.symbol$logFC), ]
table.ordered = table.ordered[table.ordered$char.GeneSymbol.fdr.005 != 'NA', ]
table.ordered = table.ordered[table.ordered$adj.P.Val < 0.001, ]

adj.P.Val = table.ordered$adj.P.Val[1:50]
symbols = table.ordered$char.GeneSymbol.fdr.005[1:50]
logFC = table.ordered$logFC[1:50]

top_genes = as.data.frame(cbind(logFC, adj.P.Val, symbols))

top_genes
```

```
##                logFC            adj.P.Val  symbols
## 1  -3.16910596282394 3.03974958361597e-05    ANXA3
## 2  -2.90196354594275 5.33542743683499e-06    DHRS2
## 3  -2.69066220909813 5.58722806825158e-06   CD300A
## 4   -2.4783691581818 5.33542743683499e-06     HES4
## 5   -2.4376179001124 6.39763917300571e-05 MIR210HG
## 6  -2.42314361311855 4.28762601132574e-05    BCAT1
## 7  -2.35353059401076 9.24995023932772e-06  SLC16A6
## 8  -2.26914497545292 8.43124281768207e-06      AK4
## 9  -2.22261025680651 0.000149270583074668      HK2
## 10  -2.1574682382529 9.24995023932772e-06     DTX1
## 11 -2.11789035211356 8.71449498813624e-05    DOCK5
## 12 -2.09323757950462  2.0211404995319e-05    BCAT1
## 13 -2.03811243013619 8.66193463662085e-06  BHLHE40
## 14 -1.93957440033025 1.73626086573341e-05     HES1
## 15 -1.93915716300933 1.26503107251277e-05    DDIT4
## 16 -1.93074831594866 6.29691676224944e-05     ASNS
## 17 -1.92022624744775 1.43987602151547e-05    BCAT1
## 18 -1.91464719831033 5.33542743683499e-06  MTHFD1L
## 19 -1.91413675358827 0.000108845169449817    S100P
## 20 -1.90242422780709 2.85732037314158e-05     GPT2
## 21 -1.88771446954017 9.24995023932772e-06      CR2
## 22 -1.88662320126868 0.000152892597110897     FLNA
## 23 -1.88170845963841 6.55280149610789e-05   PFKFB3
## 24 -1.87739200603505 5.33542743683499e-06   CD300A
## 25  -1.8455144167955 0.000240695339769724    VARS1
## 26 -1.83735454408087 0.000152892597110897  SLC7A11
## 27 -1.82789279465216  0.00017544961322013    SCN7A
## 28 -1.81789392714239 3.95544583580934e-05   LZTFL1
## 29 -1.76749637512118 6.55280149610789e-05     HES1
## 30 -1.75670087356455 9.24995023932772e-06     RHOU
## 31 -1.74185315789929 9.24564268027658e-05    CRACD
## 32 -1.66503618124738 4.09654546818888e-05     PDK1
## 33  -1.6567703588917 0.000207888731928294      NLN
## 34 -1.63458904423732 8.71449498813624e-05     PDK1
## 35 -1.61002469968258  0.00032390623681348    IGF1R
## 36 -1.59280694798781 9.70712155395696e-05    EPAS1
## 37 -1.59274393124748 3.10931843664976e-05     NLE1
## 38 -1.57993553948587 0.000101379267893044    VEGFA
```

```
## 39 -1.57721864485283 0.000191565180120652     FLNA
## 40 -1.57708198860543 0.000119846382251588    ELMO1
## 41 -1.56108296918842 0.000532541988918849    SUSD4
## 42 -1.55502150694909  0.00042762455888576   RPUSD3
## 43 -1.55390148719471 9.70712155395696e-05     IPO4
## 44  -1.5326989717413 6.39763917300571e-05    P4HA1
## 45 -1.52426873406581 0.000385871952582328     CD44
## 46 -1.52408812482802 0.000101379267893044  ANKRD37
## 47 -1.50736591923675 0.000178640675870782    NAA15
## 48 -1.50238702773232 9.00114847223938e-05      MPI
## 49 -1.50076353059692 0.000967743180706422    VEGFA
## 50 -1.49434106345895 0.000101575806299347   LZTFL1
```

```
table.ordered[table.ordered$char.GeneSymbol.fdr.005 == 'MYC', ]
```

```
##                 logFC AveExpr         t      P.Value    adj.P.Val        B
## 202431_s_at -1.339373 11.4403 -11.35534 6.797274e-08 6.076499e-05 8.636648
##             char.GeneSymbol.fdr.005
## 202431_s_at                     MYC
```

3. Enrichment analysis

Here we show the pathways that are modified significantly (and non-significantly) after using SHAM1.

```
rank <- toptableIQR$logFC
names(rank) <- toptable.annotated.symbol$char.GeneSymbol.fdr.005
pathways.hallmark <- gmtPathways('h.all.v7.4.symbols.gmt')

fgseaRes <- fgseaMultilevel(pathways=pathways.hallmark, stats=rank)

fgseaRes$adjPvalue <- ifelse(fgseaRes$padj <= 0.05, "significant", "non-significant")
fgseaRes = fgseaRes[order(padj, -abs(NES)), ]

pathway = fgseaRes$pathway
padj = fgseaRes$padj

fgseaRess = as.data.frame(cbind(pathway, padj))
fgseaRess
```

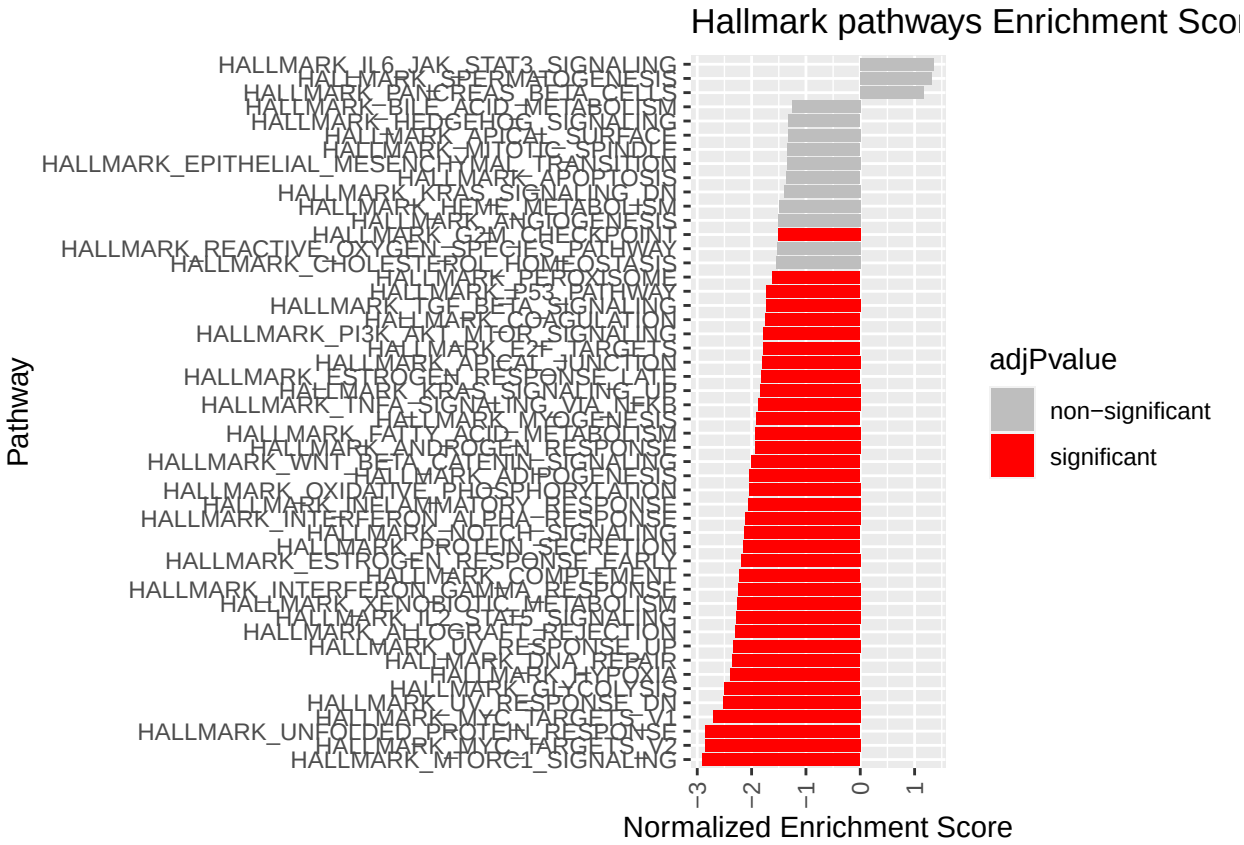
```
##                                     pathway                 padj
## 1                   HALLMARK_MTORC1_SIGNALING 8.33333333333333e-10
## 2                     HALLMARK_MYC_TARGETS_V2 8.33333333333333e-10
## 3          HALLMARK_UNFOLDED_PROTEIN_RESPONSE 8.33333333333333e-10
## 4                     HALLMARK_MYC_TARGETS_V1 8.33333333333333e-10
## 5                     HALLMARK_UV_RESPONSE_DN 8.33333333333333e-10
## 6                         HALLMARK_GLYCOLYSIS 8.33333333333333e-10
## 7                            HALLMARK_HYPOXIA 3.91567865754096e-08
## 8                HALLMARK_ALLOGRAFT_REJECTION 1.97286422073025e-07
## 9                     HALLMARK_UV_RESPONSE_UP 3.05931233982261e-07
## 10         HALLMARK_INTERFERON_GAMMA_RESPONSE 3.05931233982261e-07
## 11               HALLMARK_IL2_STAT5_SIGNALING 3.60538193689738e-07
## 12                        HALLMARK_DNA_REPAIR 9.76008480816602e-07
## 13                        HALLMARK_COMPLEMENT 1.28428644610233e-06
```

```
## 14           HALLMARK_ESTROGEN_RESPONSE_EARLY 3.55504388165894e-06
## 15             HALLMARK_XENOBIOTIC_METABOLISM 5.40614015675573e-06
## 16                   HALLMARK_NOTCH_SIGNALING  2.6527219822541e-05
## 17         HALLMARK_INTERFERON_ALPHA_RESPONSE 4.11963372933863e-05
## 18                      HALLMARK_ADIPOGENESIS 4.11963372933863e-05
## 19             HALLMARK_INFLAMMATORY_RESPONSE 4.88184555469572e-05
## 20                 HALLMARK_PROTEIN_SECRETION 6.49693480498428e-05
## 21          HALLMARK_OXIDATIVE_PHOSPHORYLATION 0.000102112797483572
## 22        HALLMARK_WNT_BETA_CATENIN_SIGNALING 0.000454979317770139
## 23           HALLMARK_TNFA_SIGNALING_VIA_NFKB 0.000454979317770139
## 24                 HALLMARK_ANDROGEN_RESPONSE 0.000521656995403225
## 25             HALLMARK_FATTY_ACID_METABOLISM  0.00118340441755278
## 26                         HALLMARK_MYOGENESIS   0.0015662701860984
## 27                         HALLMARK_E2F_TARGETS  0.00199663447308348
## 28                   HALLMARK_APICAL_JUNCTION    0.002690823197398
## 29                  HALLMARK_KRAS_SIGNALING_UP   0.0032219686079763
## 30            HALLMARK_ESTROGEN_RESPONSE_LATE  0.00460187151155973
## 31                         HALLMARK_P53_PATHWAY  0.00637352047825703
## 32           HALLMARK_PI3K_AKT_MTOR_SIGNALING  0.00665197943553325
## 33                HALLMARK_TGF_BETA_SIGNALING  0.00783529994652994
## 34                         HALLMARK_COAGULATION   0.0168339426579753
## 35                       HALLMARK_G2M_CHECKPOINT   0.0264102657951114
## 36                          HALLMARK_PEROXISOME   0.0302327904301353
## 37   HALLMARK_REACTIVE_OXYGEN_SPECIES_PATHWAY   0.0553832521045636
## 38            HALLMARK_CHOLESTEROL_HOMEOSTASIS    0.056442509272698
## 39                   HALLMARK_HEME_METABOLISM    0.056442509272698
## 40                      HALLMARK_ANGIOGENESIS   0.0636672325976231
## 41                  HALLMARK_KRAS_SIGNALING_DN    0.108384029088875
## 42                    HALLMARK_MITOTIC_SPINDLE    0.108384029088875
## 43                          HALLMARK_APOPTOSIS    0.113532320087896
## 44 HALLMARK_EPITHELIAL_MESENCHYMAL_TRANSITION    0.128994429785987
## 45                   HALLMARK_SPERMATOGENESIS    0.128994429785987
## 46           HALLMARK_IL6_JAK_STAT3_SIGNALING    0.143708193979933
## 47                HALLMARK_HEDGEHOG_SIGNALING    0.145876325339785
## 48                    HALLMARK_APICAL_SURFACE    0.153862478777589
## 49               HALLMARK_BILE_ACID_METABOLISM    0.175021464896638
## 50                HALLMARK_PANCREAS_BETA_CELLS     0.31322505800464
```

4. Plotting the results

Finally we plot the normalized enrichment scores to visualize them better.

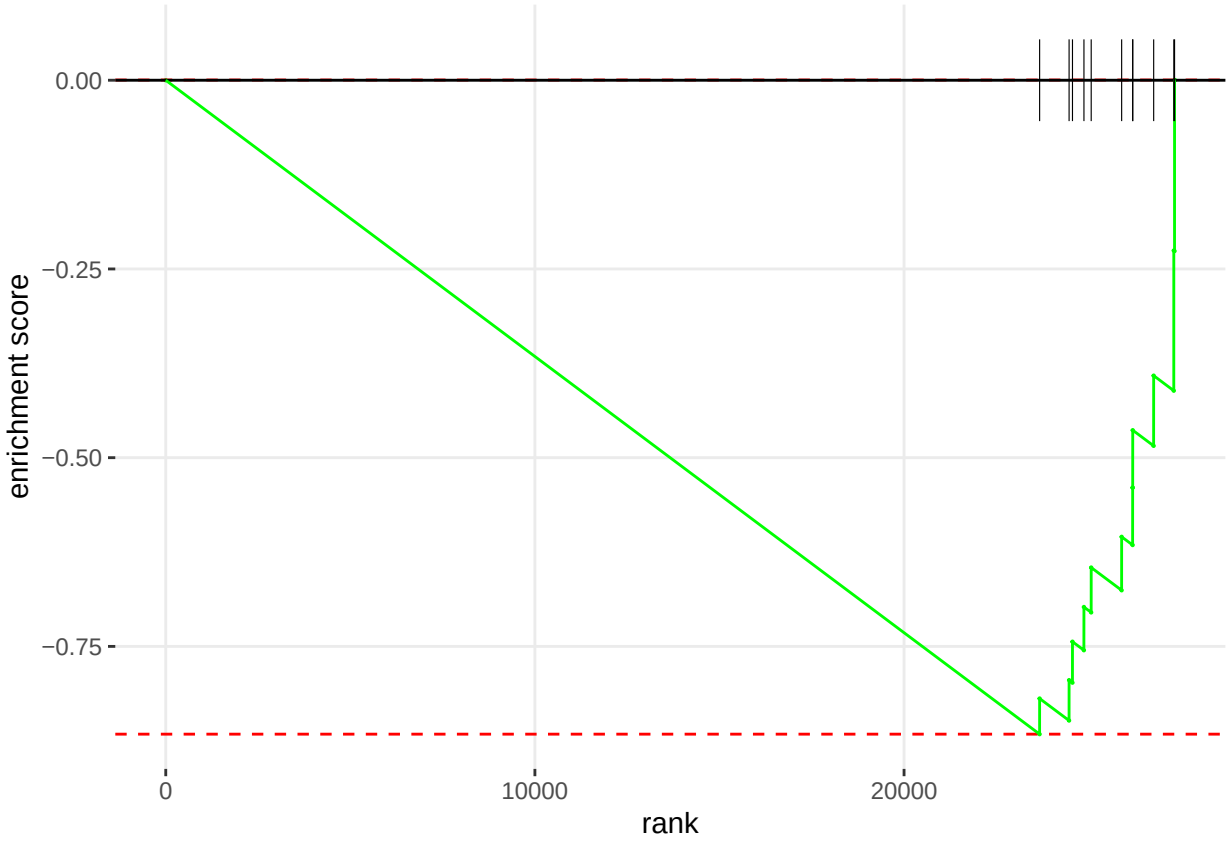
```
cols <- c("non-significant" = "grey", "significant" = "red")
library(ggplot2)
ggplot(fgseaRes, aes(reorder(pathway, NES), NES, fill = adjPvalue)) +
        geom_col() +
        scale_fill_manual(values = cols) +
        theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
        coord_flip() +
        labs(x="Pathway", y="Normalized Enrichment Score",
             title="Hallmark pathways Enrichment Score from GSEA")
```



The NOTCH signaling gene set was significantly downregulated, consistent with the findings of the original study.

Finally, we've plot the Enrichment Score for the NOTCH signaling gene set and its result is quite similar to that of the original article.

```
plotEnrichment(pathway = pathways.hallmark[["HALLMARK_NOTCH_SIGNALING"]], rank)
```


0.00
−0.25
−0.50
−0.75
enrichment score
0
10000
20000
rank