Perinatal malnutrition in male mice influences gene expression in the next generation offspring: Potential role of epigenetics.

Análisis de datos ómicos

Ana Isabel del Val

11 de abril, 2020

Contents

“

Abstract

The dataset for the exercise is available at the entry Series GSE55304 of the in Gene Expression Omnibus.

This code is already sync with ***github repository***, which is public.

This is the source reference.

Objectives

It consists on determining whether patterns of gene expression in the first generation offspring are also present in the following generation offspring, via the paternal lineage. Paternal transmission of patterns of gene expression strongly suggest epigenetic inheritance of disease risk.

Materials and methods

Data source and experiment design

Liver tissue was obtained from the following experimental groups:

- a) control male mice
- b) adult male mice previously exposed to 50% caloric restriction in utero (IUGR)

- c) adult male mice overfed during lactation (ON)
- d) adult male offspring from control mice
- e) adult male offspring from IUGR mice
- f) adult male offspring from ON mice.

RNA was extracted and processed for further hybridization on Affymetrix microarrays (GeneChip Mouse Genome 430 2.0 (Affymetrix, Santa Clara, CA)).

Targets have been created manually and is composed by 5 groups, 3 arrays in each group.

```
targets
```

```
##      FileName       Group  Genotype Treatment    ShortName
## 1  GSM1333830 BothControl      Both   Control BothControl1
## 2  GSM1333831 BothControl      Both   Control BothControl2
## 3  GSM1333832 BothControl      Both   Control BothControl3
## 4  GSM1333833   AdultIUGR     Adult      IUGR   AdultIUGR1
## 5  GSM1333834   AdultIUGR     Adult      IUGR   AdultIUGR2
## 6  GSM1333835   AdultIUGR     Adult      IUGR   AdultIUGR3
## 7  GSM1333836   AdultLact     Adult lactation   AdultLact1
## 8  GSM1333837   AdultLact     Adult lactation   AdultLact2
## 9  GSM1333838   AdultLact     Adult lactation   AdultLact3
## 10 GSM1333839     OffIUGR Offspring      IUGR     OffIUGR1
## 11 GSM1333840     OffIUGR Offspring      IUGR     OffIUGR2
## 12 GSM1333841     OffIUGR Offspring      IUGR     OffIUGR3
## 13 GSM1333842     OffLact Offspring lactation     OffLact1
## 14 GSM1333843     OffLact Offspring lactation     OffLact2
## 15 GSM1333844     OffLact Offspring lactation     OffLact3
```

Pipeline followed

1. Read data

```
head(rawData)
```

```
## ExpressionFeatureSet (storageMode: lockedEnvironment)
## assayData: 1 features, 15 samples
##   element names: exprs
## protocolData
##   rowNames: BothControl1 BothControl2 ... OffLact3 (15 total)
##   varLabels: exprs dates
##   varMetadata: labelDescription channel
## phenoData
##   rowNames: BothControl1 BothControl2 ... OffLact3 (15 total)
##   varLabels: Group Genotype Treatment ShortName
##   varMetadata: labelDescription channel
## featureData: none
## experimentData: use 'experimentData(object)'
## Annotation: pd.mouse430.2
```

2. Exploration

Quality control of raw data

The data have enough quality for normalization? If one array is above a certain threshold defined in the function it is marked with an asterisk as an outlier. When a certain array is marked three times it should be revised carefully.

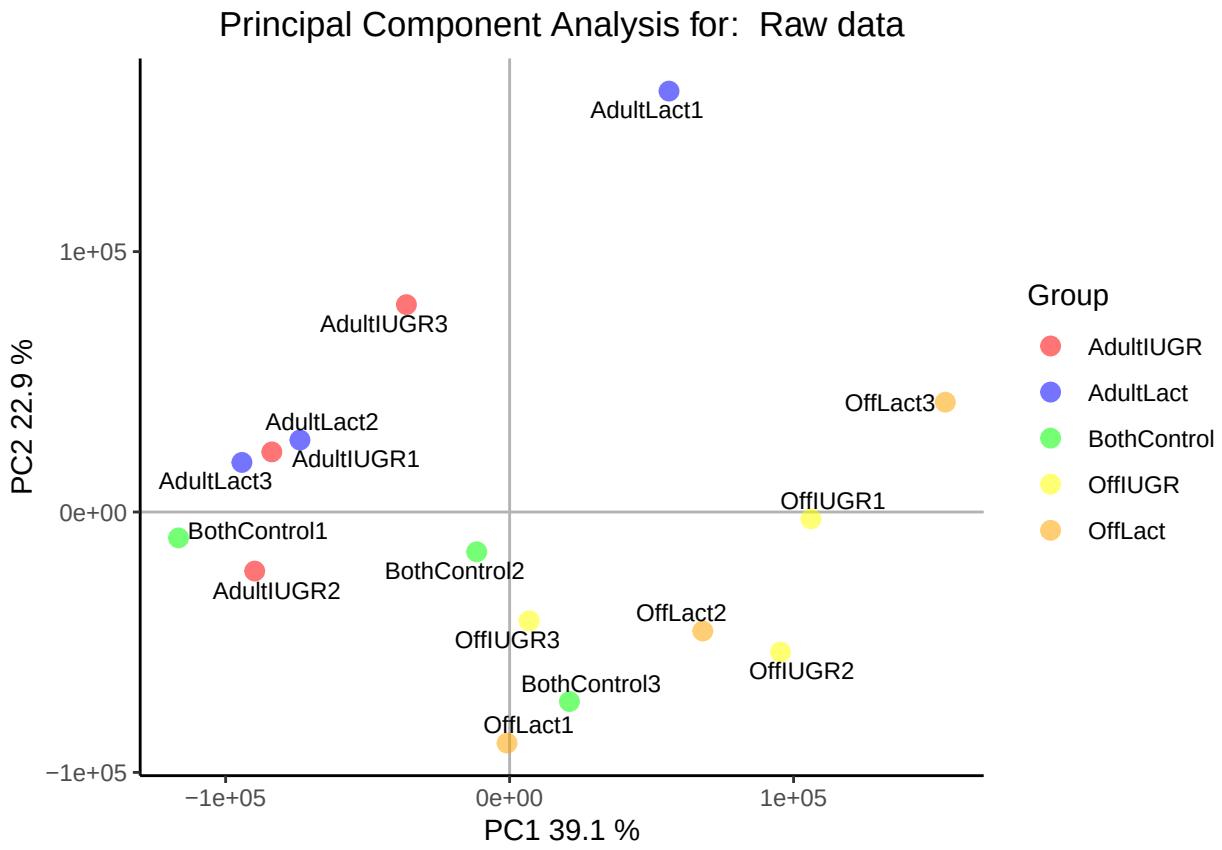
In our case, only 1 star is ticked for 3 arrays, so we don't worry about outliers.

PCA

5 colors are needed to represent this scatterplot of the first two principal components performed on the raw data.

First component of the PCA accounts for 39.1% of the total variability of the samples, and as we can observe in the plot, this variability is mainly contributed by the sample generation, as offsprings are on the right and adults are on the left, except for the array AdultLact1.

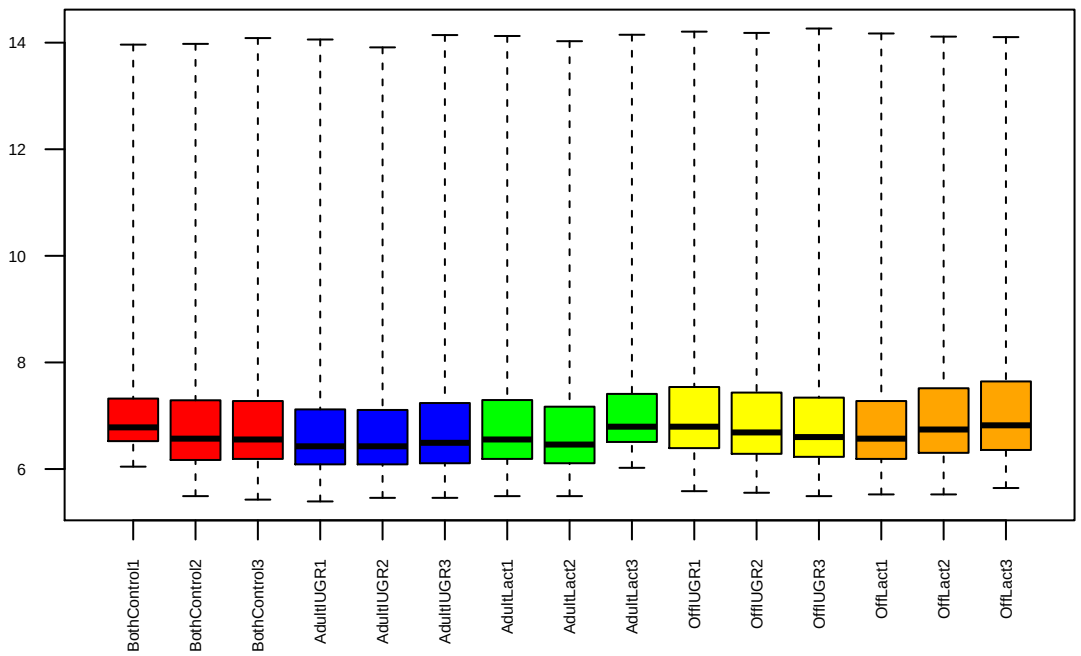
```
plotPCA3(exprs(rawData), labels = targets$ShortName, factor = targets$Group,
title="Raw data", scale = FALSE, size = 3,
colores = c("red", "blue", "green", "yellow", "orange"))
```



We save image to tiff file in figures folder.

With a boxplot we visualize the intensity distribution of the arrays. The group legend colors are different in boxplot and PCA, so let's have a deep look before interpretation.

```
boxplot(rawData, cex.axis=0.5, las=2,  which="all",
col = c(rep("red", 3), rep("blue", 3),
        rep("green", 3), rep("yellow", 3), rep("orange", 3))
,main="Distribution of raw intensity values")
```



A light variation of intensity among arrays is observed, but this is the expected for raw data.

Data normalization

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

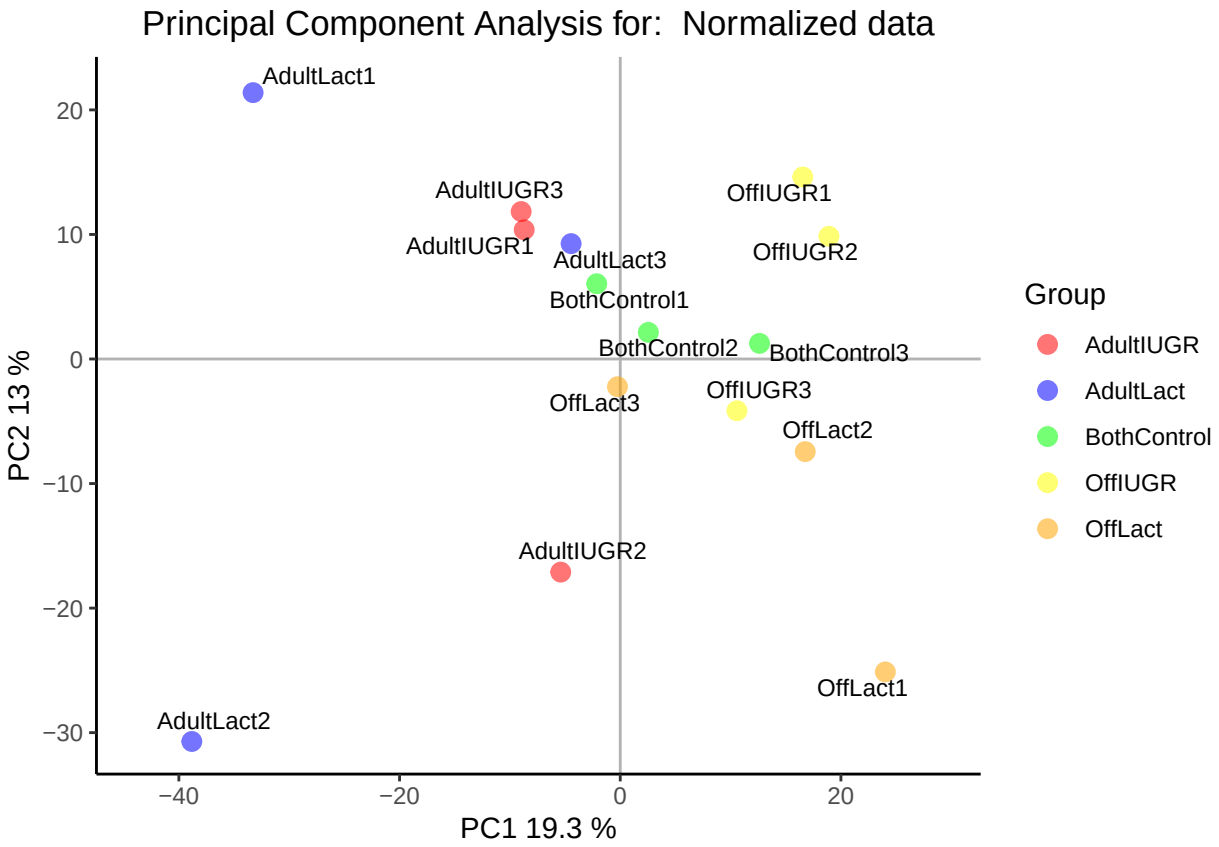
```
## Background correcting
## Normalizing
## Calculating Expression
```

Quality control of normalized data

After normalization, the array AdultLact1 has been moved to the left part of the scatterplot.

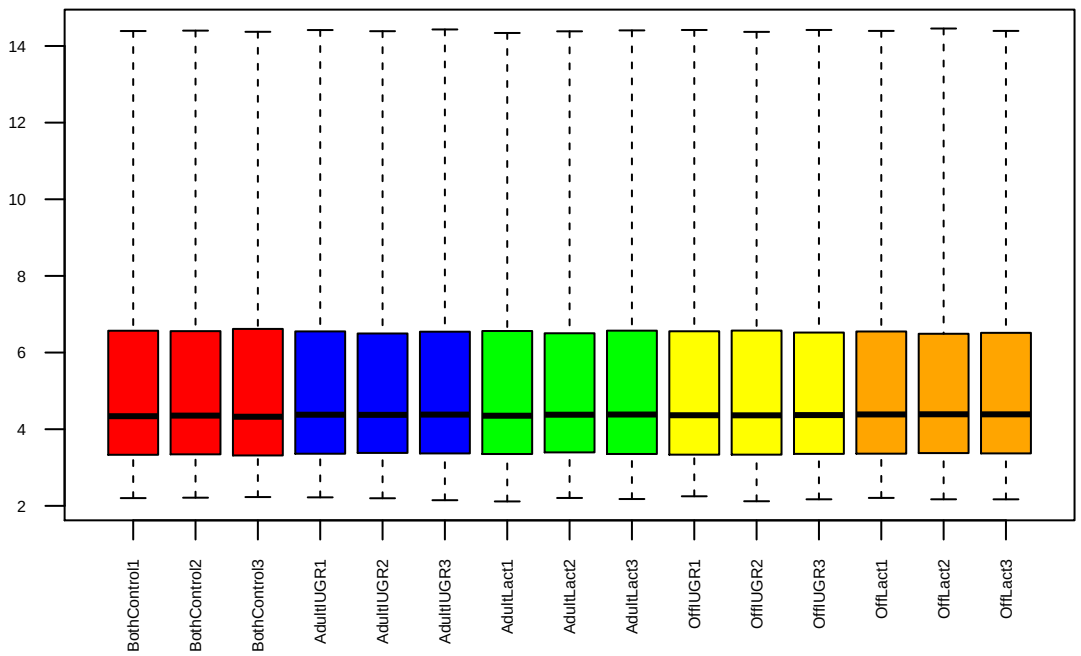
First component of the PCA accounts for 19.3% of the total variability. It separates samples by the generation, as offsprings are on the right and adults are on the left. After normalization, without exception.

```
plotPCA3(exprs(eset_rma), labels = targets$ShortName, factor = targets$Group,
title="Normalized data", scale = FALSE, size = 3,
colores = c("red", "blue", "green", "yellow","orange"))
```



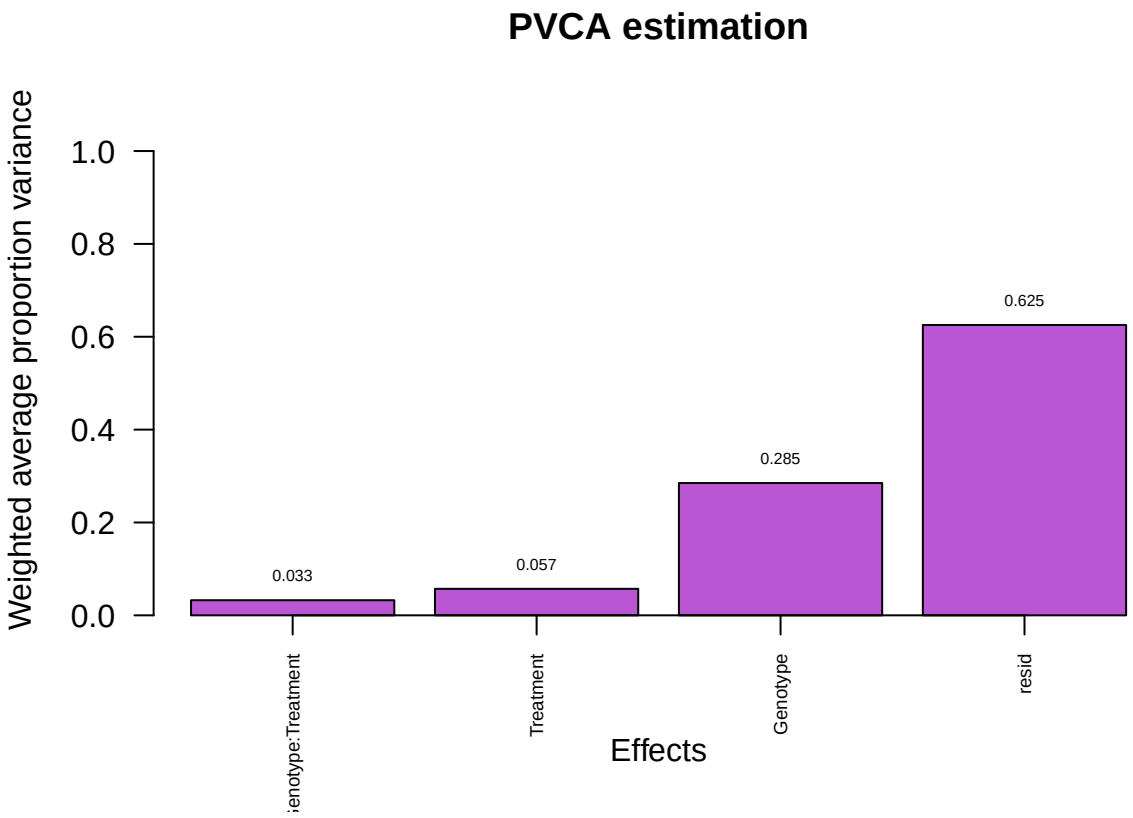

```
boxplot(eset_rma, cex.axis=0.5, las=2,  which="all",
col = c(rep("red", 3), rep("blue", 3), rep("green", 3), rep("yellow", 3),rep("orange", 3)),
main="Boxplot for arrays intensity: Normalized Data")
```

```
## Warning in .local(x, ...): Argument 'which' ignored (not meaningful for ExpressionSet)
```



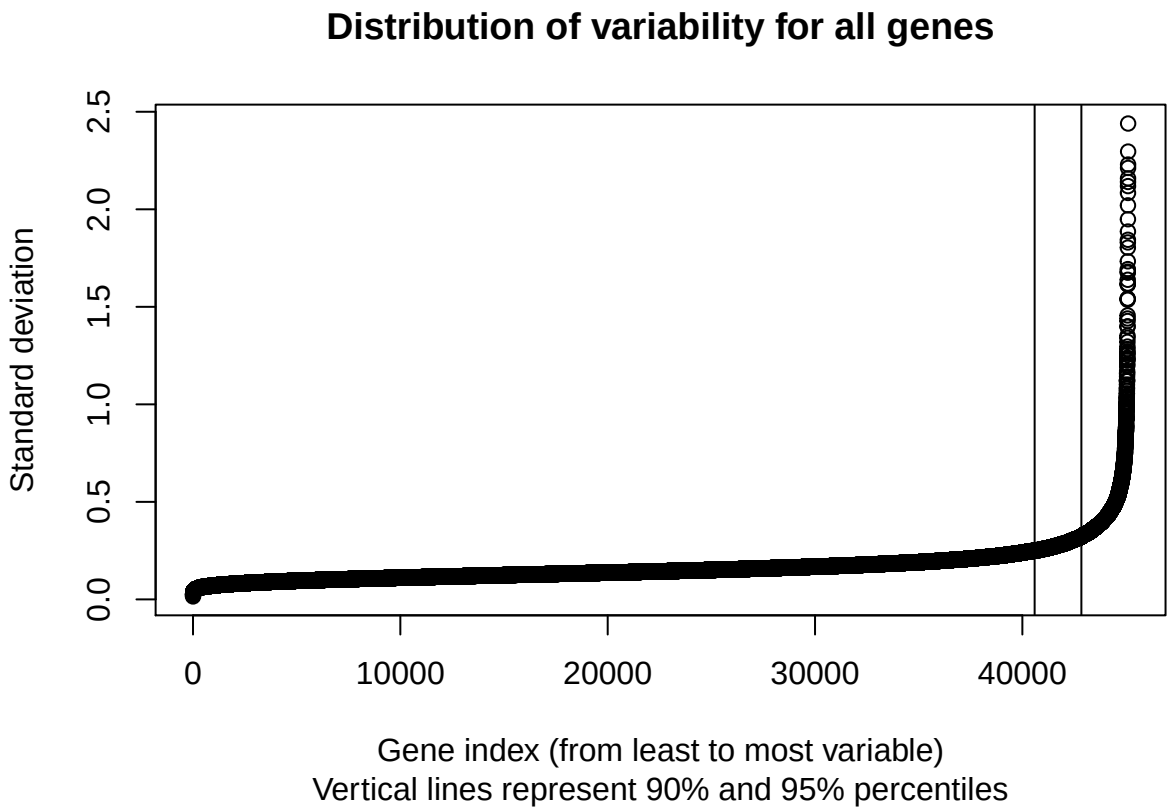
Batch detection

```
bp <- barplot(pvcaObj$dat, xlab = "Effects",
ylab = "Weighted average proportion variance",
ylim= c(0,1.1),col = c("mediumorchid"), las=2,
main="PVCA estimation")

axis(1, at = bp, labels = pvcaObj$label, cex.axis = 0.55, las=2)

values = pvcaObj$dat

new_values = round(values , 3)

text(bp,pvcaObj$dat,labels = new_values, pos=3, cex = 0.5)
```



Detecting most variable genes

```
sds <- apply (exprs(eset_rma), 1, sd)
sds0<- sort(sds)
plot(1:length(sds0), sds0, main="Distribution of variability for all 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))
```



Values of standard deviations allong all samples for all genes ordered from smallest to biggest

Filter least variable genes

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

```
## $numDupsRemoved
## [1] 17422
##
## $numLowVar
## [1] 15416
##
## $numRemoved.ENTREZID
## [1] 7111
##
## $feature.exclude
## [1] 13
```

Before filtering, there were 45101 genes.

After filtering, there are 5139 genes left.

Save normalized data

We save data in results folder.

3. Define the experimental setup

Compare gene expression between groups.

The Linear Models for Microarrays method, implemented in the limma package Smyth (2005) is used to select differential expressed genes.

Create the design matrix

The first step for the analysis based on linear models is to create the design matrix. Basically it is a table that describes the allocation of each sample to a group or experimental condition. It has as many rows as samples and as many columns as groups (if only one factor is considered). Each row contains a one in the column of the group to which the sample belongs and a zero in the others.

1 model of 1 factor with 5 levels defined in Targets>Groups > 5 columns.

```
designMat<- model.matrix(~0+Group, pData(eset_filtered))
colnames(designMat) <- c("BothControl", "AdultIUGR", "AdultLact", "OffIUGR","OffLact")
```

Defining comparisons with the Contrasts Matrix

It consists of as many columns as comparisons and as many rows as groups (that is, as columns of the design matrix) -> (5rows by 3 columns).

A comparison between groups - called "contrast" - is represented by a "1" and a "-1" in the rows of groups to compare and zeros in the rest.

3 comparisons > 3 columns in the contrast matrix.

Build the contrast matrix that can be used to answer the following questions:

- Compare the effect of IntraUterine Growth Restriction in offsprings.
- Compare the effect of overfed during lactation in offsprings.
- The interaction: the differences between the two previous effects in offsprings.

There could be more comparisons to be made, but I have highlighted here which I consider the most interesting ones.

```
cont.matrix <- makeContrasts (BothControlvsOffIUGR = BothControl-OffIUGR,
                              BothControlvsOffLact = BothControl-OffLact,
                              INT = OffIUGR - OffLact,levels=designMat)
print(cont.matrix)
```

```
##              Contrasts
## Levels        BothControlvsOffIUGR BothControlvsOffLact INT
##   BothControl                    1                    1   0
##   AdultIUGR                      0                    0   0
##   AdultLact                      0                    0   0
##   OffIUGR                       -1                    0   1
##   OffLact                        0                   -1  -1
```

Model estimation and gene selection

With LIMMA, once the design matrix and the contrasts have been defined, we can proceed to estimate the model, estimate the contrasts and perform the significance tests that will lead to the decision, for each gene and each comparison, if they can be considered differential expressed.

The analysis provides the usual test statistics such as Fold-change t-moderated or adjusted p-values that are used to order the genes from more unless differential expressed.

In order to control the percentage of false positives that may result from high number of contrasts made simultaneously the p-values are adjusted so that we have control over the false positive rate using the Benjamini and Hochberg method Benjamini and Hochberg (1995).

topTable: for a given contrast a list of genes ordered from smallest to biggest p–value which can be considered to be most to least differential expressed.

For Comparison 1:

```
topTab_BothControlvsOffIUGR <-
  topTable(fit.main,
           number=nrow(fit.main),
           coef="BothControlvsOffIUGR",
           adjust="fdr")
head(topTab_BothControlvsOffIUGR)
```

```
##                 logFC  AveExpr         t      P.Value   adj.P.Val         B
## 1438758_at -2.3282725 7.522021 -7.667917 1.420024e-06 0.007297503 4.6330862
## 1450064_at -0.9691825 5.919838 -5.544024 5.565138e-05 0.123768511 1.8148609
## 1439059_at -1.3650712 3.503159 -5.351001 8.006620e-05 0.123768511 1.5192822
## 1417600_at  0.9120025 6.368973  5.207271 1.053043e-04 0.123768511 1.2950501
## 1415685_at -0.6306643 8.318089 -5.026904 1.490648e-04 0.123768511 1.0088038
## 1440771_at -1.6306961 5.828590 -4.991487 1.596670e-04 0.123768511 0.9519765
```

For Comparison 2:

```
topTab_BothControlvsOffLact <-
  topTable(fit.main,
           number=nrow(fit.main),
           coef="BothControlvsOffLact",
           adjust="fdr")
head(topTab_BothControlvsOffLact)
```

```
##                   logFC  AveExpr         t      P.Value  adj.P.Val        B
## 1431182_at   -2.1237194 6.235149 -6.270831 1.479835e-05 0.03736404 3.074858
## 1421061_at    0.7838334 4.975084  6.084998 2.062107e-05 0.03736404 2.797971
## 1418243_at   -1.0436027 7.227763 -5.890252 2.934492e-05 0.03736404 2.501370
## 1420634_a_at  1.3477017 6.054184  5.888456 2.944129e-05 0.03736404 2.498604
## 1427357_at    1.2081038 7.753155  5.773408 3.635341e-05 0.03736404 2.320273
## 1433966_x_at -2.4863124 4.070717 -5.651317 4.556100e-05 0.03902299 2.128548
```

For Comparison 3:

```
topTab_INT <-
  topTable(fit.main,
  number=nrow(fit.main),
  coef="INT",
  adjust="fdr")
head(topTab_INT)
```

```
##                  logFC   AveExpr         t      P.Value  adj.P.Val        B
## 1438758_at    2.091071  7.522021  6.886719 5.097879e-06 0.02619800 3.652820
## 1421225_a_at  1.033863  8.182224  6.153676 1.823124e-05 0.04684518 2.678164
## 1428636_at   -1.541085  5.162313 -5.658011 4.499814e-05 0.05895101 1.964637
## 1424041_s_at -1.001887 11.290039 -5.589442 5.112300e-05 0.05895101 1.862482
## 1460467_at    1.151154  5.162744  5.512947 5.898825e-05 0.05895101 1.747545
## 1424351_at   -1.497486  7.095931 -5.430945 6.882780e-05 0.05895101 1.623196
```

First column of each topTable contains the manufacturer's (Affymetrix) ID for each probeset. Next step is to guess which gene correspond to each Affymetrix ID. This process is called annotation. Gene Symbol, the Entrez Gene identifier or the Gene description.

Annotation tables, one per comparison.

Let's see for the first comparison:

```
short_BothControlvsOffIUGR <- head(topAnnotated_BothControlvsOffIUGR[1:5,1:4])
short_BothControlvsOffIUGR
```

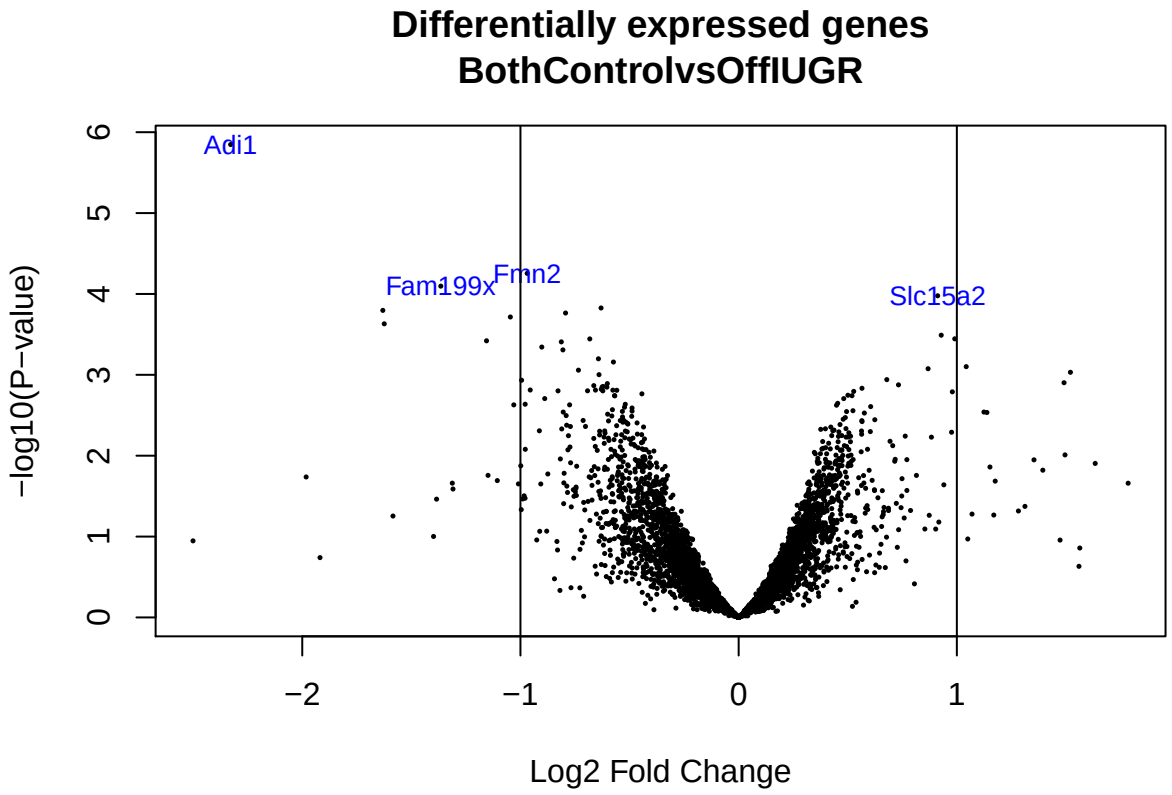
```
##      PROBEID SYMBOL ENTREZID                                        GENENAME
## 1 1415673_at   Psph   100678                        phosphoserine phosphatase
## 2 1415685_at  Mtif2    76784 mitochondrial translational initiation factor 2
## 3 1415698_at  Golm1   105348                         golgi membrane protein 1
## 4 1415743_at  Hdac5    15184                            histone deacetylase 5
## 5 1415750_at   Tbl3   213773                         transducin (beta)-like 3
```

Visualizating differential expression

The names of the top 4 genes are shown in blue in the figure.

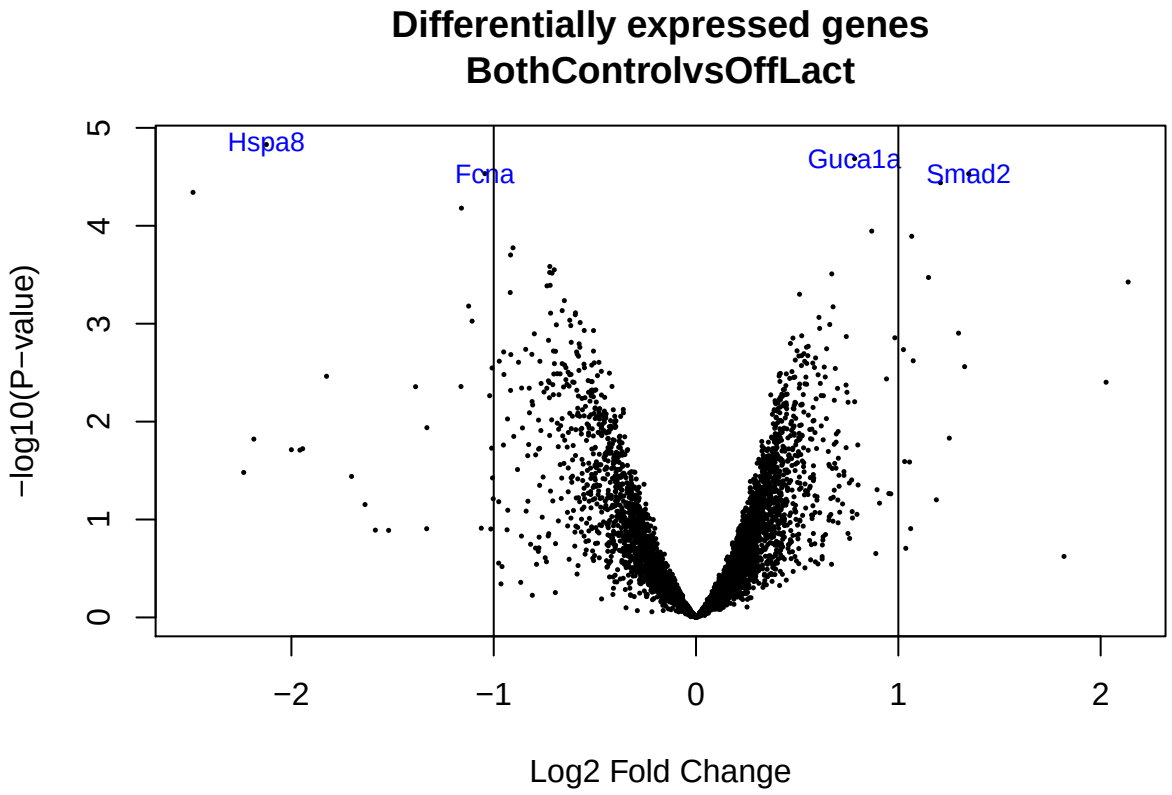
This is for BothControlvsOffIUGR comparison.

```
volcanoplot(fit.main, coef=1, highlight=4, names=SYMBOLS,
main=paste("Differentially expressed genes", colnames(cont.matrix)[1], sep="\n"))
abline(v=c(-1,1))
```



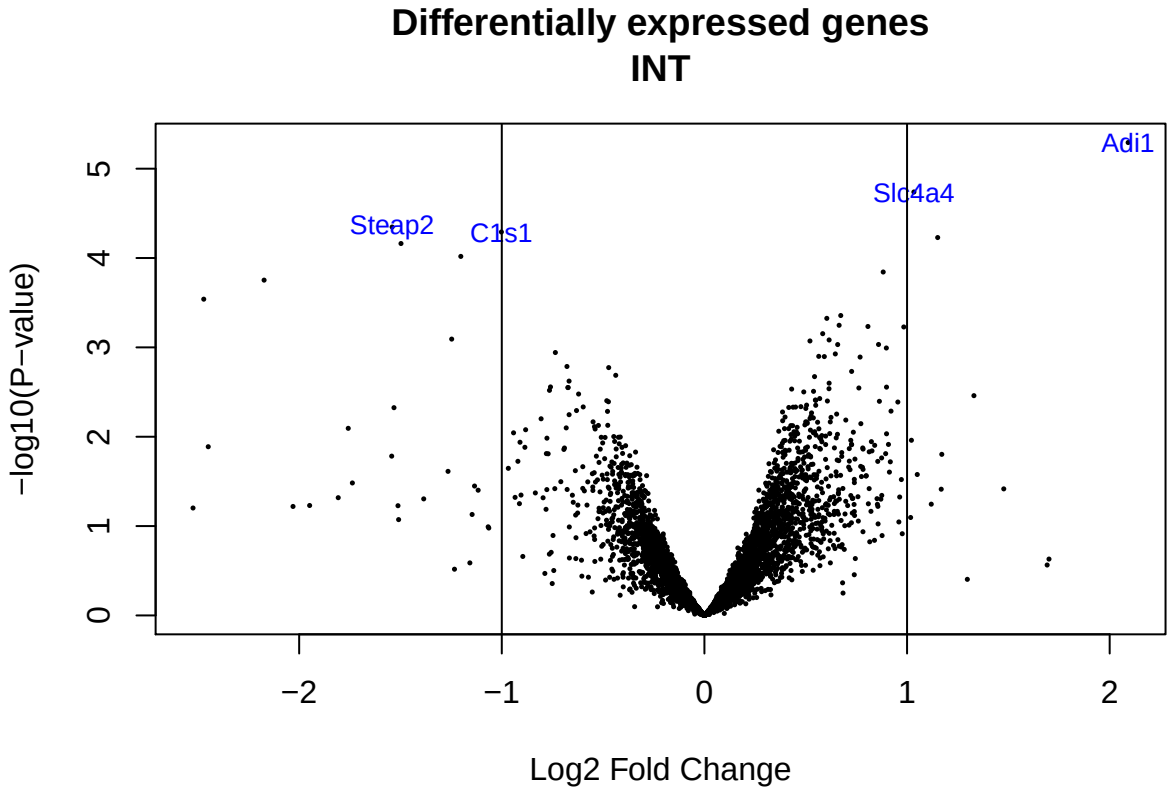
For second comparison BothControlvsOffLact:

```
volcanoplot(fit.main, coef=2, highlight=4, names=SYMBOLS,
main=paste("Differentially expressed genes", colnames(cont.matrix)[2], sep="\n"))
abline(v=c(-1,1))
```



For third comparison INT:

```
volcanoplot(fit.main, coef=3, highlight=4, names=SYMBOLS,
main=paste("Differentially expressed genes", colnames(cont.matrix)[3], sep="\n"))
abline(v=c(-1,1))
```



Multiple comparisons

When one selects genes in several comparisons it is usually interesting to know which genes have been selected in each comparison. Sometimes biologically relevant genes will be those that are selected in one of them but not in others. In other occasions he interest will lie in genes that are selected in all comparisons.

This object has as many columns as comparisons and as many rows as genes: 5139x3.

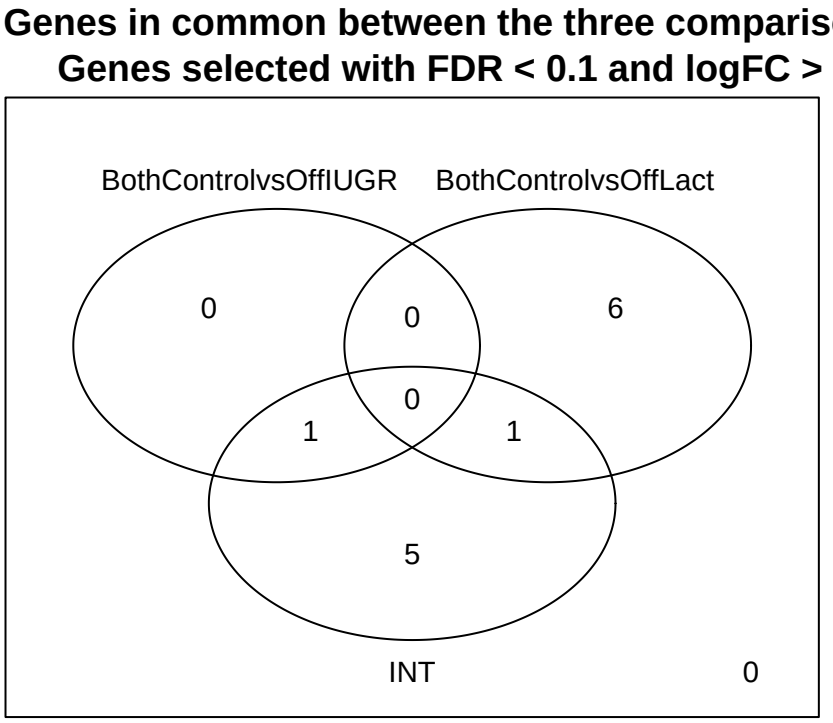
Per each gene and comparison a "+1" denotes significantly up-regulated (t-test values >0, FDR < selected cutoff), a "-1" significantly down-regulated (t-test values <0, FDR < selected cutoff) and a "0" non significant difference (FDR > selected cutoff).

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

```
##        BothControlvsOffIUGR BothControlvsOffLact  INT
## Down                      1                    4    4
## NotSig                 5138                 5132 5132
## Up                        0                    3    3
```

This can be visualized in a Venn Diagram.

```
vennDiagram (res.selected[,1:3], cex=0.9)
title("Genes in common between the three comparisons\n Genes selected with FDR < 0.1 and logFC > 1")
```



Venn diagram showing the genes in common between the three comparisons performed.

4. Expression profiles visualization: Heatmaps

Genes that have been selected as differential expressed may be visualized using a heatmap. These plots use color palettes to highlight distinct values –here positive (up-regulation) or negative (down-regulation) significantly differential expressions.

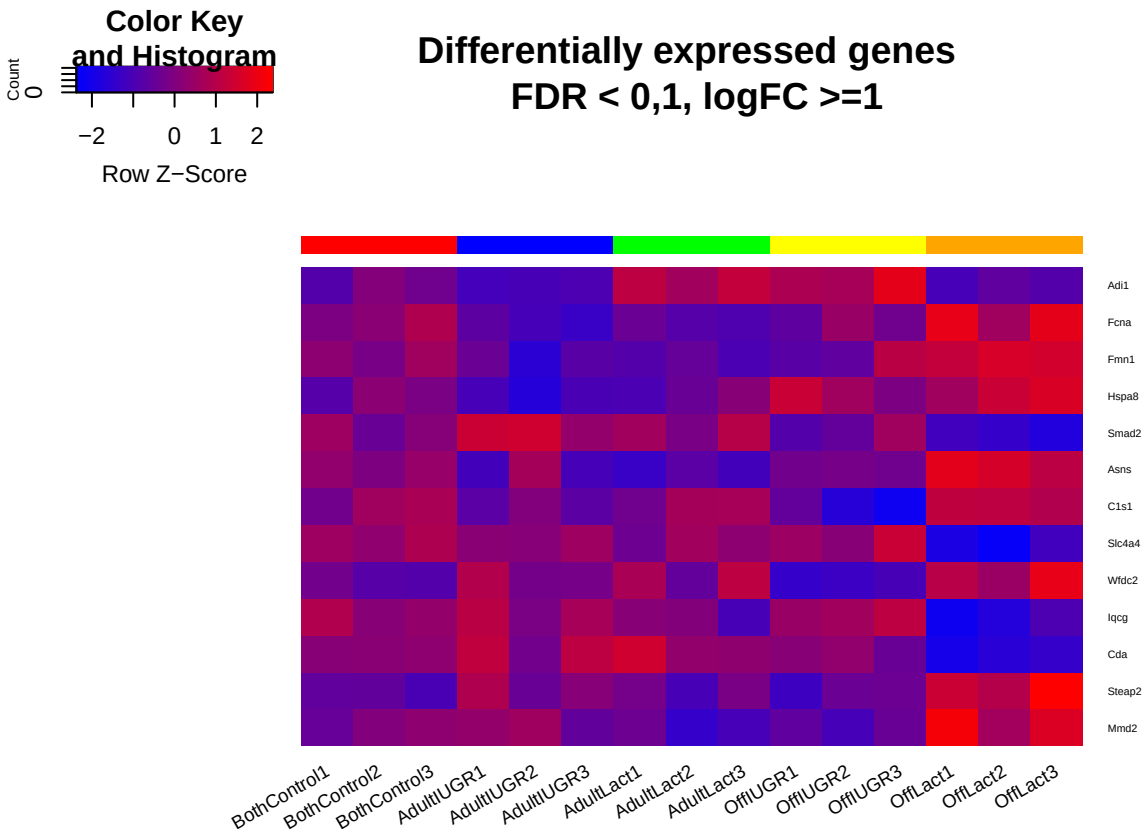
Heatmaps can be used to visualize the expression values of differential expressed genes with no specific order, but it is usually preferred to plot them doing a hierarchical clustering on genes (rows) or columns(samples) in order to find groups of genes with common patterns of variation which can eventually be associated to the different groups being compared.

A common option is to select the gens that have been selected in the previous steps, that is the genes that have been called differential expressed in at least one comparison.

(FDR < 0.1 and logFC > 1)

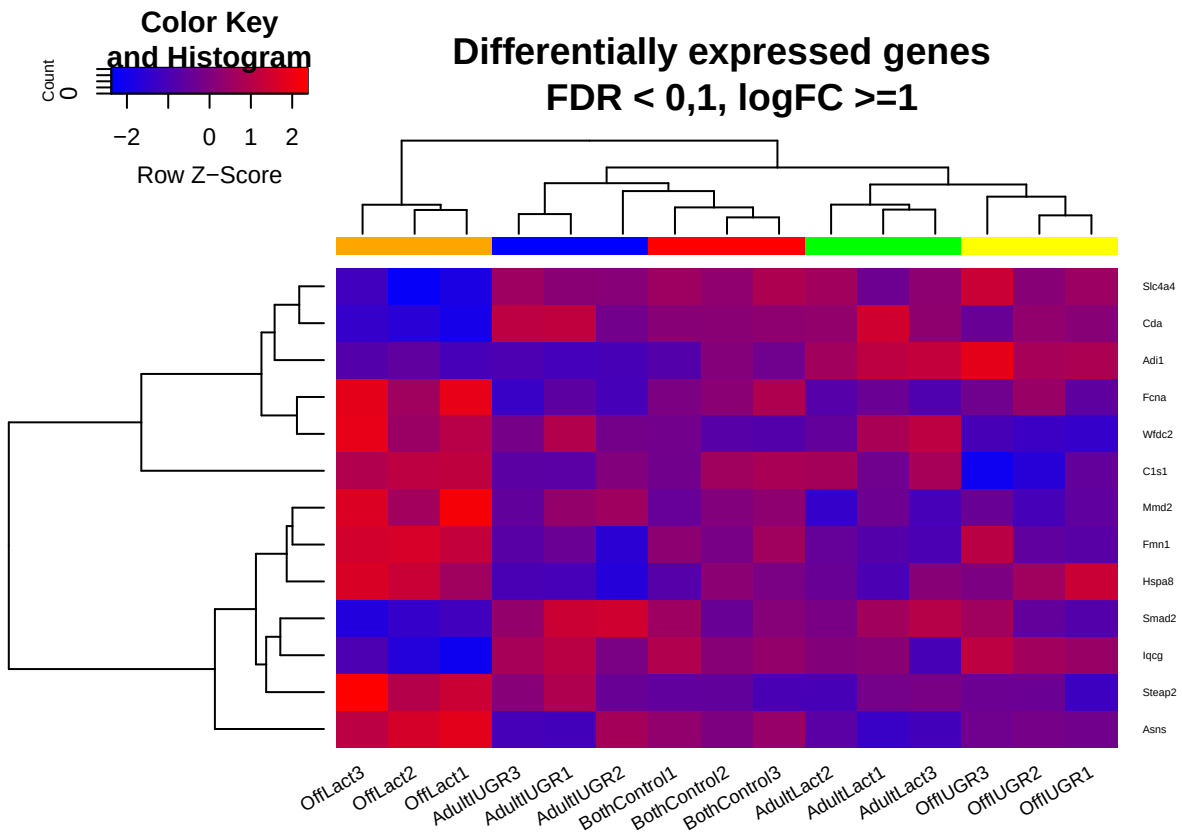
```
my_palette <- colorRampPalette(c("blue", "red"))(n = 299)

heatmap.2(HMdata,
Rowv = FALSE,
Colv = FALSE,
main = "Differentially expressed genes \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",3),rep("blue",3), rep("green",3), rep("yellow",3),rep("orange",3)),
tracecol = NULL,
dendrogram = "none",
srtCol = 30)
```



Genes and samples are forced to group by row and column similarity respectively.

```
heatmap.2(HMdata,
Rowv = TRUE,
Colv = TRUE,
dendrogram="both",
main = "Differentially expressed genes \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",3),rep("blue",3), rep("green",3), rep("yellow",3),rep("orange",3)),
tracecol = NULL,
srtCol = 30)
```



Biological Significance of results

Given a list of genes selected for being differential expressed between two conditions, the functions, biological processes or molecular pathways that characterize them appear on this list more frequently than among the rest of the genes analyzed.

ReactomePA Bioconductor package. The analysis is done on the ReactomePA annotation database https://reactome.org/.

Analyses of this type need a minimum number of genes to be reliable, preferably a few hundreds than a few dozens, so it is common to perform a selection less restrictive than with the previous steps. For instance an option is to include all genes with a non-stringent FDR cutoff, such as FDR < 0.15 without filtering by minimum "fold-change").

The analysis also requires to have the Entrez Identifiers for all genes analyzed. It is an open discussion if what one should use is "all genes analyzed" -that is genes that have been retained in the analysis and are part of the "topTable"- or all genes available. In this case we use the second option and define our universe to be all genes that have at least one annotation in the Gene Ontology.

The Biological significance analysis will be applied to these lists: "BothControlvsOffLact" "INT".

First rows and columns for Reactome results on INT.csv comparison:

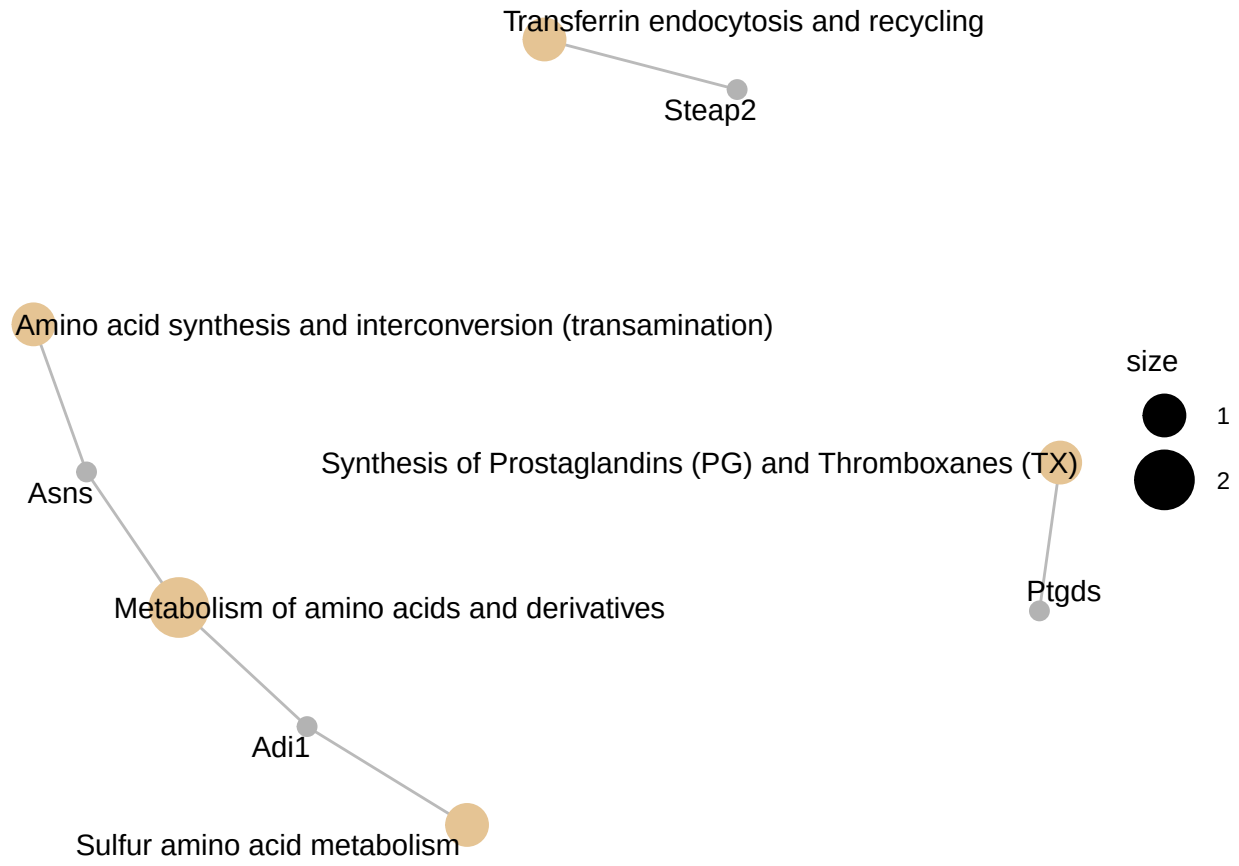
```
Tab.react <- read.csv2(file.path("./results/ReactomePA.Results.INT.csv"), sep = ",", header = TRUE, row.nam

Tab.react <- Tab.react[1:4, 1:4]
knitr::kable(Tab.react, booktabs = TRUE, caption = "First rows and columns for Reactome results on INT.csv
```

This netwowrk figure shows the network produced from the genes selected in the comparison

Table 1: First rows and columns for Reactome results on INT.csv comparison

	Description	GeneRatio	BgRatio	pvalue
R-MMU-71291	Metabolism of amino acids and derivatives	2/5	200/8698	0.005025769852516
R-MMU-2162123	Synthesis of Prostaglandins (PG) and Thromboxanes (TX)	1/5	12/8698	0.006880707975511
R-MMU-1614635	Sulfur amino acid metabolism	1/5	20/8698	0.011446766067409
R-MMU-917977	Transferrin endocytosis and recycling	1/5	31/8698	0.017697657633436

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



In comparison INT, 8 enriched pathways have been found, for example: Sulfur amino acid metabolism.

The results obtained in the analysis of biological significance are:

- a .csv file with a summary of all the enriched pathways and the associated statistics.
- a bar plot with the best enriched pathways. Height of the bar plot is the number of genes of our analysis related with that pathway. Moreover, pathways are ordered by statistical significance.
- a plot with a network of the enriched pathways and the relation among the genes included.

Discussion

I have found a limitation in comparison BothControlvsOffIUGR. There was not enriched pathway found so Reactome results on this comparison was NULL.

For future iterations of this study, we could think of other comparisons such as:

- Compare the effect of IntraUterine Growth Restriction in adults.
- Compare the interaction of IntraUterine Growth Restriction in adults and offsprings.
- Compare the effect of overfed during lactation in adults.
- Compare the interaction of overfed during lactation in adults and offsprings.

Apendix

The code can be found in the Github repository, in .RMd file, where the reproducibility of the study is guaranteed.