732A51 Bioinformatics

LAB 4 Bioinformatics

Hugo Morvan
William Wiik

STIMA
Department of Computer and Information Science
Linköpings universitet

2024-12-08

Contents

```r
if (!require("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
BiocManager::install()
BiocManager::install("GEOquery")
library(GEOquery)

#ERROR: dependencies 'affy', 'genefilter', 'gcrma' are not available for package 'simpleaffy'
BiocManager::install('affy')
BiocManager::install('affyPLM')
BiocManager::install('genefilter')
BiocManager::install('gcrma')
install.packages('./simpleaffy_2.50.0.tar.gz', type='source') #Needs Linux ?
library(simpleaffy)

library(RColorBrewer)

library(limma)

BiocManager::install('hgu133plus2.db')
library(hgu133plus2.db)

library(annotate)

library(ggplot2)
```

1 Question 1

Run all the R code and reproduce the graphics. Go carefully through the R code and explain in your words what each step does. HINT Recall what a design/model matrix is from linear regression.

Loading the GEO data using `GEOquery` package. Extracting the downloaded raw `.tar` files into data directory. Creating the phenotype data by generating a matrix and then converting it do a data frame. Manually adding `Targets` labels. Saves the phenotype data.

```r
# Important note: before knitting, delete the data folder
library(GEOquery)
x = getGEOSuppFiles("GSE20986")
x
```

```
##                                                                          size
## /home/h/Documents/LiU/Bioinformatics/Lab4/GSE20986/GSE20986_RAW.tar 56360960
##                                                                       isdir mode
## /home/h/Documents/LiU/Bioinformatics/Lab4/GSE20986/GSE20986_RAW.tar FALSE  664
##                                                                               mtime
## /home/h/Documents/LiU/Bioinformatics/Lab4/GSE20986/GSE20986_RAW.tar 2024-12-05 09:29:07
##                                                                               ctime
## /home/h/Documents/LiU/Bioinformatics/Lab4/GSE20986/GSE20986_RAW.tar 2024-12-05 09:29:07
```

```
##                                                                                 atime
## /home/h/Documents/LiU/Bioinformatics/Lab4/GSE20986/GSE20986_RAW.tar 2024-12-05 09:28:14
##                                                                                  uid  gid
## /home/h/Documents/LiU/Bioinformatics/Lab4/GSE20986/GSE20986_RAW.tar 1000 1000
##                                                                                uname
## /home/h/Documents/LiU/Bioinformatics/Lab4/GSE20986/GSE20986_RAW.tar     h
##                                                                               grname
## /home/h/Documents/LiU/Bioinformatics/Lab4/GSE20986/GSE20986_RAW.tar      h
```

```
untar("GSE20986/GSE20986_RAW.tar", exdir = "data")
cels = list.files("data/", pattern = "[gz]")
sapply(paste("data", cels, sep = "/"), gunzip)
```

```
## data/GSM524662.CEL.gz data/GSM524663.CEL.gz data/GSM524664.CEL.gz
##              13555726              13555055              13555639
## data/GSM524665.CEL.gz data/GSM524666.CEL.gz data/GSM524667.CEL.gz
##              13560122              13555663              13557614
## data/GSM524668.CEL.gz data/GSM524669.CEL.gz data/GSM524670.CEL.gz
##              13556090              13560054              13555971
## data/GSM524671.CEL.gz data/GSM524672.CEL.gz data/GSM524673.CEL.gz
##              13554926              13555042              13555290
```

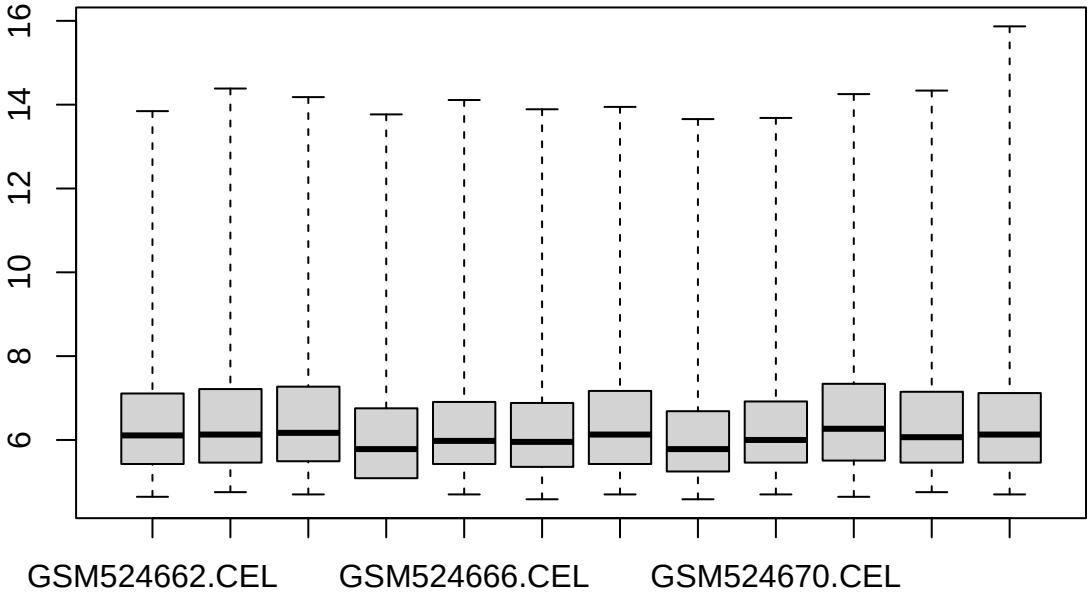
```
phenodata = matrix(rep(list.files("data"), 2), ncol =2)
class(phenodata)
```

```
## [1] "matrix" "array"
```

```
phenodata <- as.data.frame(phenodata)
colnames(phenodata) <- c("Name", "FileName")
phenodata$Targets <- c("iris",
                       "retina",
                       "retina",
                       "iris",
                       "retina",
                       "iris",
                       "choroid",
                       "choroid",
                       "choroid",
                       "huvec",
                       "huvec",
                       "huvec")
write.table(phenodata, "data/phenodata.txt", quote = F, sep = "\t", row.names = F)
```

The `read.affy()` function loads the CEL files into an AffyBatch object. Then making a boxplot to show the distribution of raw intensity values for each samples.

```
library(simpleaffy)
celfiles <- read.affy(covdesc = "phenodata.txt", path = "data")
boxplot(celfiles)
```

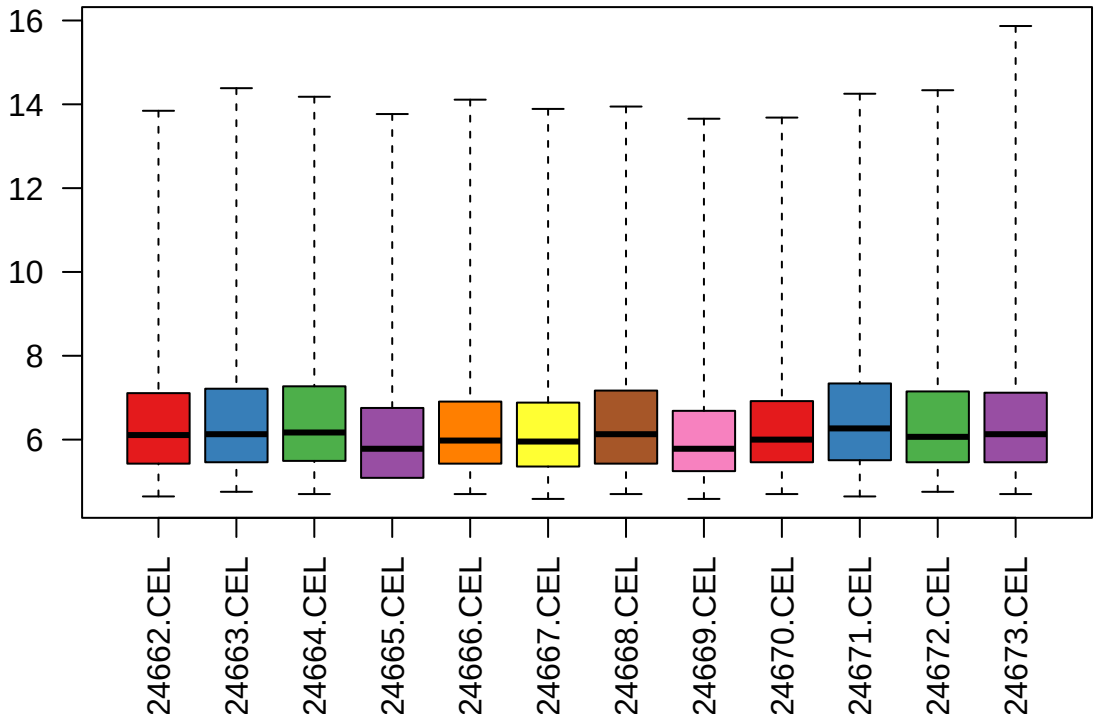


Extracting raw expression values (eset) from CEL files and name columns using sample labels. Then creating a boxplot of raw expression values with colors for better visualization.

```
library(RColorBrewer)
cols = brewer.pal(8, "Set1")
eset <- exprs(celfiles)
samples <- celfiles$Targets
colnames(eset)
```

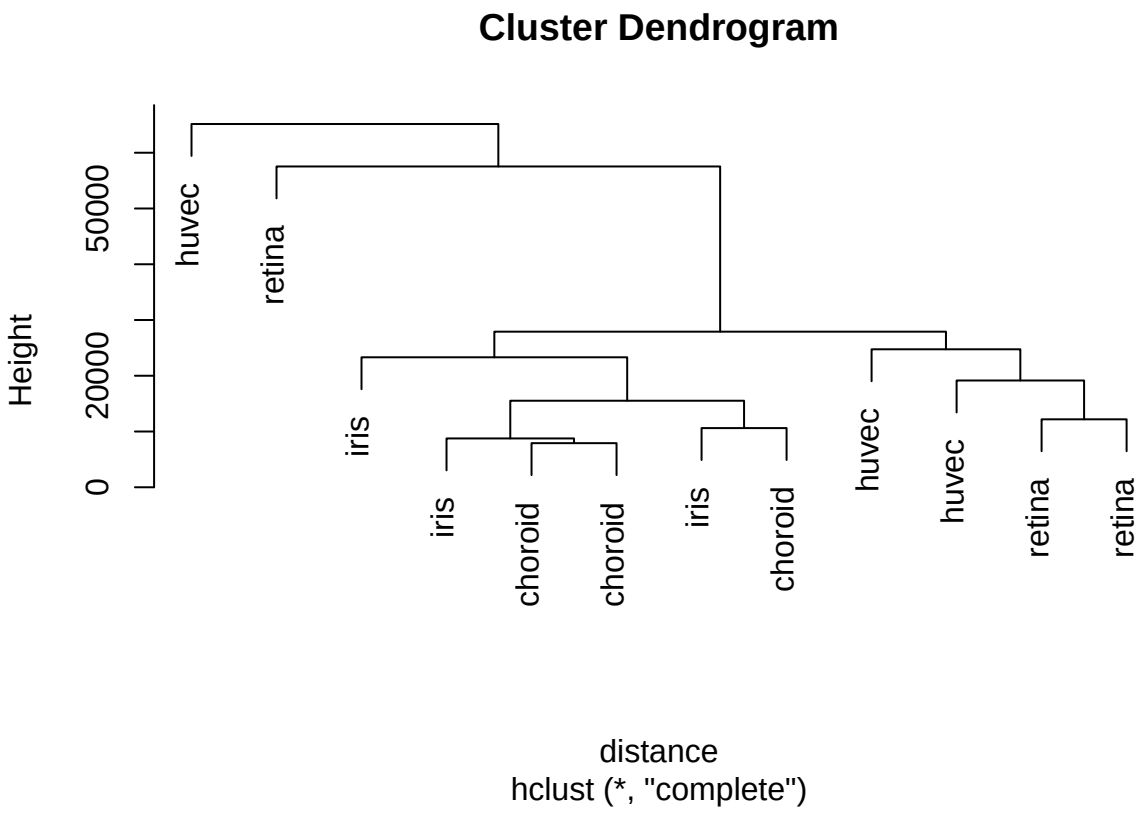
```
##  [1] "GSM524662.CEL" "GSM524663.CEL" "GSM524664.CEL" "GSM524665.CEL"
##  [5] "GSM524666.CEL" "GSM524667.CEL" "GSM524668.CEL" "GSM524669.CEL"
##  [9] "GSM524670.CEL" "GSM524671.CEL" "GSM524672.CEL" "GSM524673.CEL"
```

```
colnames(eset) <- samples
boxplot(celfiles, col = cols, las = 2)
```



Computes pairwise distances between samples using the maximum distance metric. Doing hierarchical clustering and visualize it as a dendrogram to group samples by similarity.

```
distance <- dist(t(eset), method = "maximum")
clusters <- hclust(distance)
plot(clusters)
```



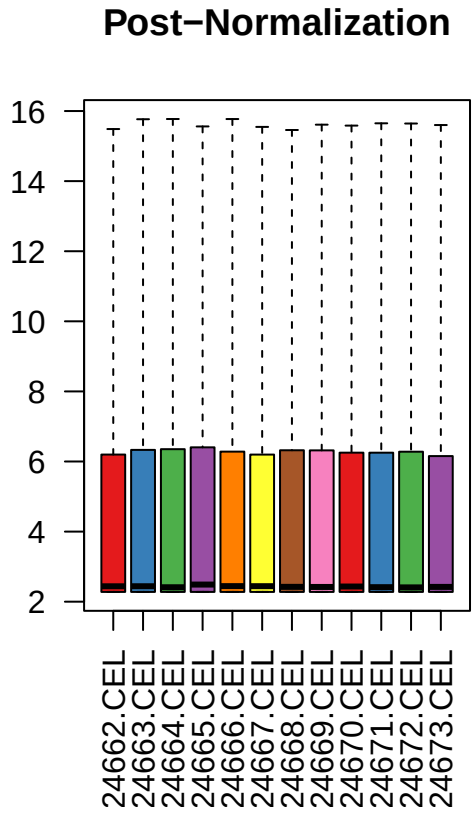

```
require(simpleaffy)
require(affyPLM)
```

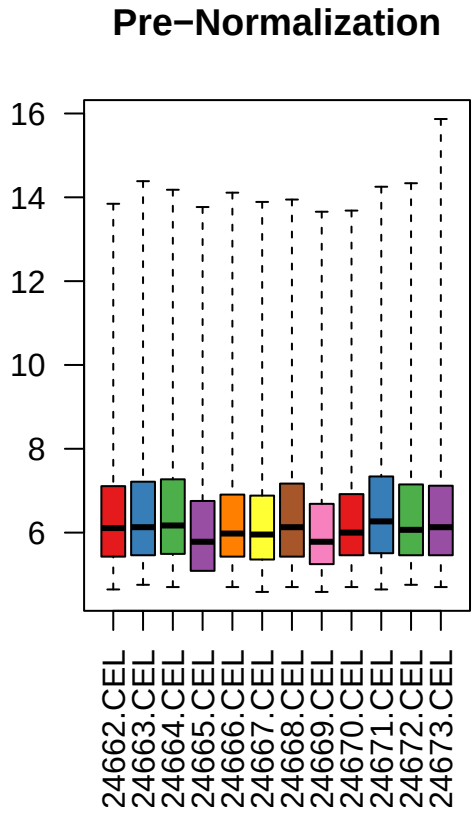
Normalizing raw data using `gcrma()`. Then plotting two boxplots to compare the data before and after the normalization.

```
celfiles.gcrma = gcrma(celfiles)
```

```
## Adjusting for optical effect.....Done.
## Computing affinities.Done.
## Adjusting for non-specific binding.....Done.
## Normalizing
## Calculating Expression
```

```
par(mfrow=c(1,2))
boxplot(celfiles.gcrma, col = cols, las = 2, main = "Post-Normalization");
boxplot(celfiles, col = cols, las = 2, main = "Pre-Normalization")
```






```
#dev.off()
library(limma)
```

Converting the class labels to factors and create a design matrix for the linear model (rows = samples, columns = groups). Specifying contrast for comparison between groups.

```
phenodata
```

```
##             Name      FileName Targets
## 1  GSM524662.CEL GSM524662.CEL    iris
## 2  GSM524663.CEL GSM524663.CEL  retina
## 3  GSM524664.CEL GSM524664.CEL  retina
## 4  GSM524665.CEL GSM524665.CEL    iris
## 5  GSM524666.CEL GSM524666.CEL  retina
## 6  GSM524667.CEL GSM524667.CEL    iris
## 7  GSM524668.CEL GSM524668.CEL choroid
## 8  GSM524669.CEL GSM524669.CEL choroid
## 9  GSM524670.CEL GSM524670.CEL choroid
## 10 GSM524671.CEL GSM524671.CEL   huvec
## 11 GSM524672.CEL GSM524672.CEL   huvec
## 12 GSM524673.CEL GSM524673.CEL   huvec
```

```
samples <- as.factor(samples)
design <- model.matrix(~0+samples)
colnames(design)
```

```
## [1] "sampleschoroid" "sampleshuvec"   "samplesiris"    "samplesretina"
```

```
colnames(design) <- c("choroid", "huvec", "iris", "retina")
design
```

```
##    choroid huvec iris retina
## 1        0     0    1      0
## 2        0     0    0      1
## 3        0     0    0      1
## 4        0     0    1      0
## 5        0     0    0      1
## 6        0     0    1      0
## 7        1     0    0      0
## 8        1     0    0      0
## 9        1     0    0      0
## 10       0     1    0      0
## 11       0     1    0      0
## 12       0     1    0      0
## attr(,"assign")
## [1] 1 1 1 1
## attr(,"contrasts")
## attr(,"contrasts")$samples
## [1] "contr.treatment"
```

```
contrast.matrix = makeContrasts(
              huvec_choroid = huvec - choroid,
              huvec_retina = huvec - retina,
              huvec_iris = huvec - iris,
              levels = design)
```

```
library(hgu133plus2.db)

library(annotate)
```

Fitting the linear model using `lmFit()` and then computing the contrast from the linear model. Using `eBayes()` to get empirical Bayes statistics for analysis. Filter results based on statistical thresholds for significance and log fold change, then categorize into three classes, upregulated, downregulated, or non-significant.

```
fit = lmFit(celfiles.gcrma, design)
huvec_fit <- contrasts.fit(fit, contrast.matrix)
huvec_ebay <- eBayes(huvec_fit)
```

```
probenames.list <- rownames(topTable(huvec_ebay, number = 100000))
getsymbols <- getSYMBOL(probenames.list, "hgu133plus2")
results <- topTable(huvec_ebay, number = 100000, coef = "huvec_choroid")
results <- cbind(results, getsymbols)

summary(results)
```

```
##      logFC              AveExpr             t                 P.Value
##  Min.   :-9.19178   Min.   : 2.279   Min.   :-39.77095   Min.   :0.0000
##  1st Qu.:-0.05972   1st Qu.: 2.281   1st Qu.: -0.70703   1st Qu.:0.1522
##  Median : 0.00000   Median : 2.480   Median :  0.00000   Median :0.5080
##  Mean   :-0.02355   Mean   : 4.375   Mean   :  0.07445   Mean   :0.5345
##  3rd Qu.: 0.03970   3rd Qu.: 6.241   3rd Qu.:  0.67369   3rd Qu.:1.0000
##  Max.   : 8.66974   Max.   :15.542   Max.   :295.37719   Max.   :1.0000
##    adj.P.Val            B           getsymbols
##  Min.   :0.0000   Min.   :-7.711   Length:54675
##  1st Qu.:0.6036   1st Qu.:-7.711   Class :character
##  Median :1.0000   Median :-7.452   Mode  :character
##  Mean   :0.7436   Mean   :-6.583
##  3rd Qu.:1.0000   3rd Qu.:-6.498
##  Max.   :1.0000   Max.   :21.296
```

```
results$threshold <- "1"
a <- subset(results, adj.P.Val < 0.05 & logFC > 5)
results[rownames(a), "threshold"] <- "2"
b <- subset(results, adj.P.Val < 0.05 & logFC < -5)
results[rownames(b), "threshold"] <- "3"
table(results$threshold)
```

```
##
##     1     2     3
## 54587    33    55
```

Creating a Volcano Plot using `ggplot()` with log fold change (logFC) on the x-axis and adjusted p-values on the y-axis (log-transformed). Color points by threshold categories (non-significant, upregulated, downregulated). Annotate genes with high upregulation and significance.

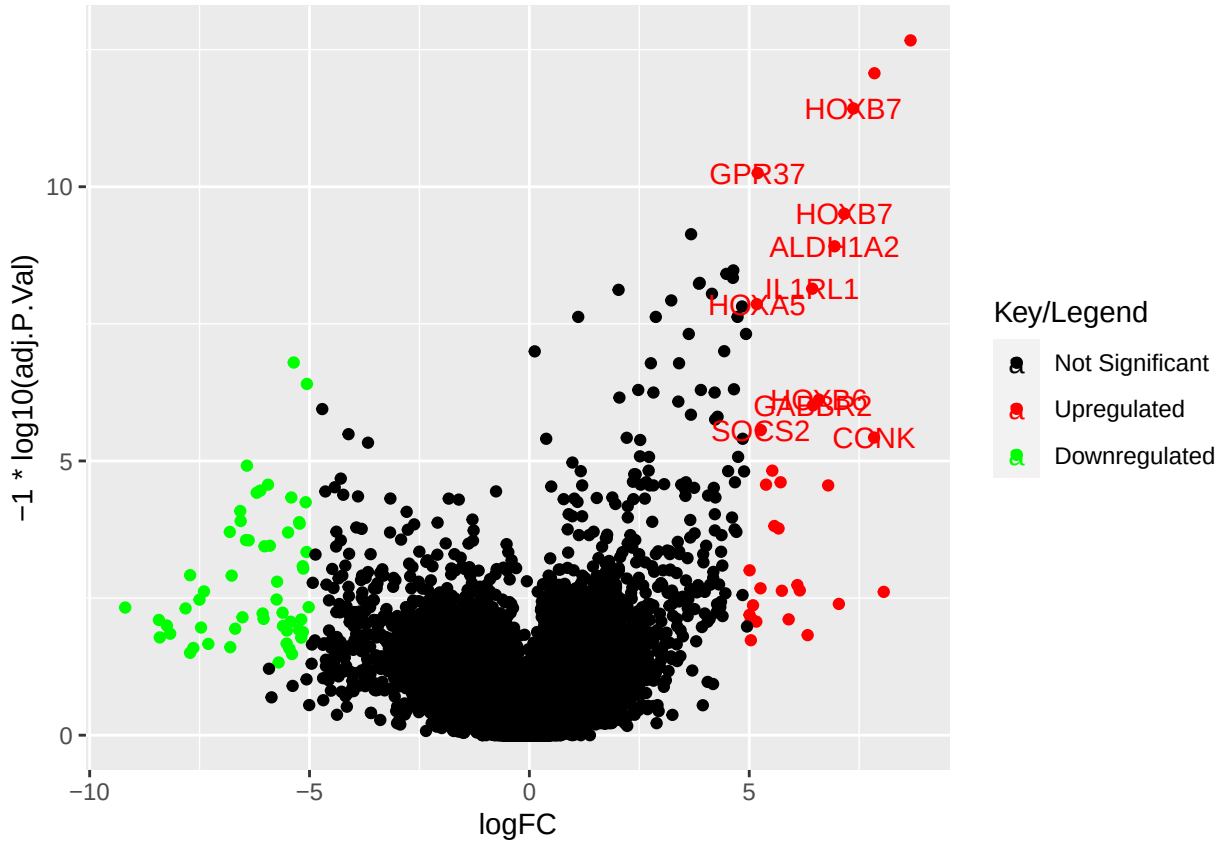
```
library(ggplot2)
volcano <- ggplot(data = results,
                  aes(x = logFC, y = -1*log10(adj.P.Val),
                      colour = threshold,
                      label = getsymbols))

volcano <- volcano +
  geom_point() +
  scale_color_manual(values = c("black", "red", "green"),
                     labels = c("Not Significant", "Upregulated", "Downregulated"),
```

```
                    name = "Key/Legend")

volcano +
  geom_text(data = subset(results, logFC > 5 & -1*log10(adj.P.Val) > 5), aes(x = logFC, y = -1*log10(adj
```

```
## Warning: Removed 2 rows containing missing values (`geom_text()`).
```



2 Question 2

In the presented analysis, there are no plots of raw paired data. In the section where the contrasts are defined find the three contrasts. Present the variables versus each other original, log–scaled and MA–plot for each considered pair both before and after normalization. A cluster analysis is performed on the page but not reported. Present plots and also draw heatmaps.

2.1 huvec_choroid pair

```
# Original

# Log-scaled

# MA-plot
```

2.2 huvec_iris pair

```
# Original

# Log-scaled

# MA-plot
```

2.3 huvec_retina pair

```
# Original

# Log-scaled

# MA-plot
```

2.4 cluster analysis ?

3 Question 3

The volcano plot is only for huvec versus choroid. Provide volcano plots for the other pairs. Indicate significantly differentially expressed genes. Explain how they are found.

3.1 Huvec - Retina pair volcano plot

```
results <- topTable(huvec_ebay, number = 100000, coef = "huvec_retina")
results <- cbind(results, getsymbols)

summary(results)
```

```
##      logFC              AveExpr             t                P.Value
##  Min.   :-10.97732   Min.   : 2.279   Min.   :-40.22393   Min.   :0.0000
##  1st Qu.: -0.05040   1st Qu.: 2.281   1st Qu.: -0.60808   1st Qu.:0.1564
##  Median :  0.00000   Median : 2.480   Median :  0.00000   Median :0.5236
##  Mean   : -0.03206   Mean   : 4.375   Mean   :  0.08633   Mean   :0.5414
##  3rd Qu.:  0.04602   3rd Qu.: 6.241   3rd Qu.:  0.72407   3rd Qu.:1.0000
##  Max.   :  8.66974   Max.   :15.542   Max.   :295.37719   Max.   :1.0000
##    adj.P.Val            B           getsymbols
##  Min.   :0.0000   Min.   :-7.711   Length:54675
##  1st Qu.:0.6127   1st Qu.:-7.711   Class :character
##  Median :1.0000   Median :-7.470   Mode  :character
##  Mean   :0.7525   Mean   :-6.615
##  3rd Qu.:1.0000   3rd Qu.:-6.523
##  Max.   :1.0000   Max.   :21.295
```

```
results$threshold <- "1"
a <- subset(results, adj.P.Val < 0.05 & logFC > 5)
results[rownames(a), "threshold"] <- "2"
b <- subset(results, adj.P.Val < 0.05 & logFC < -5)
results[rownames(b), "threshold"] <- "3"
table(results$threshold)
```

```
##
##     1     2     3
## 54557    24    94
```

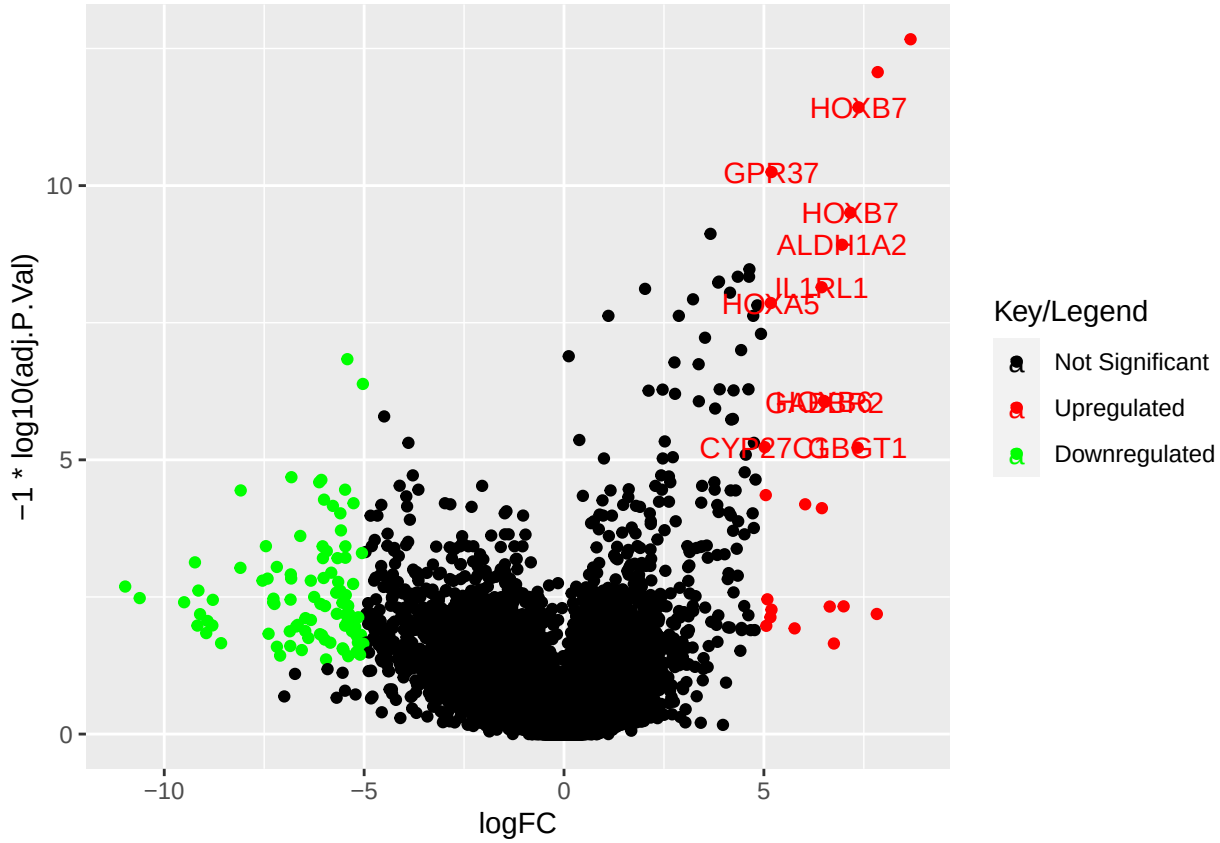
```
volcano <- ggplot(data = results,
                  aes(x = logFC, y = -1*log10(adj.P.Val),
                      colour = threshold,
                      label = getsymbols))

volcano <- volcano +
  geom_point() +
  scale_color_manual(values = c("black", "red", "green"),
```

```
                    labels = c("Not Significant", "Upregulated", "Downregulated"),
                    name = "Key/Legend")

volcano +
  geom_text(data = subset(results, logFC > 5 & -1*log10(adj.P.Val) > 5), aes(x = logFC, y = -1*log10(adj
```

```
## Warning: Removed 2 rows containing missing values (`geom_text()`).
```



3.2 Huvec - Iris pair volcano plot

```
results <- topTable(huvec_ebay, number = 100000, coef = "huvec_iris")
results <- cbind(results, getsymbols)

summary(results)
```

```
##      logFC              AveExpr              t                 P.Value
##  Min.   :-8.26522   Min.   : 2.279   Min.   :-42.51679   Min.   :0.0000
```

```
##  1st Qu.:-0.08707   1st Qu.: 2.281   1st Qu.: -1.14357   1st Qu.:0.1252
##  Median : 0.00000   Median : 2.480   Median :  0.00000   Median :0.3679
##  Mean   :-0.02250   Mean   : 4.375   Mean   : -0.00822   Mean   :0.4888
##  3rd Qu.: 0.03889   3rd Qu.: 6.241   3rd Qu.:  0.66191   3rd Qu.:1.0000
##  Max.   : 8.66974   Max.   :15.542   Max.   :295.37719   Max.   :1.0000
##    adj.P.Val            B           getsymbols
##  Min.   :0.0000   Min.   :-7.711   Length:54675
##  1st Qu.:0.5007   1st Qu.:-7.711   Class :character
##  Median :0.7358   Median :-7.231   Mode  :character
##  Mean   :0.6797   Mean   :-6.441
##  3rd Qu.:1.0000   3rd Qu.:-6.320
##  Max.   :1.0000   Max.   :21.295
```

```
results$threshold <- "1"
a <- subset(results, adj.P.Val < 0.05 & logFC > 5)
results[rownames(a), "threshold"] <- "2"
b <- subset(results, adj.P.Val < 0.05 & logFC < -5)
results[rownames(b), "threshold"] <- "3"
table(results$threshold)
```

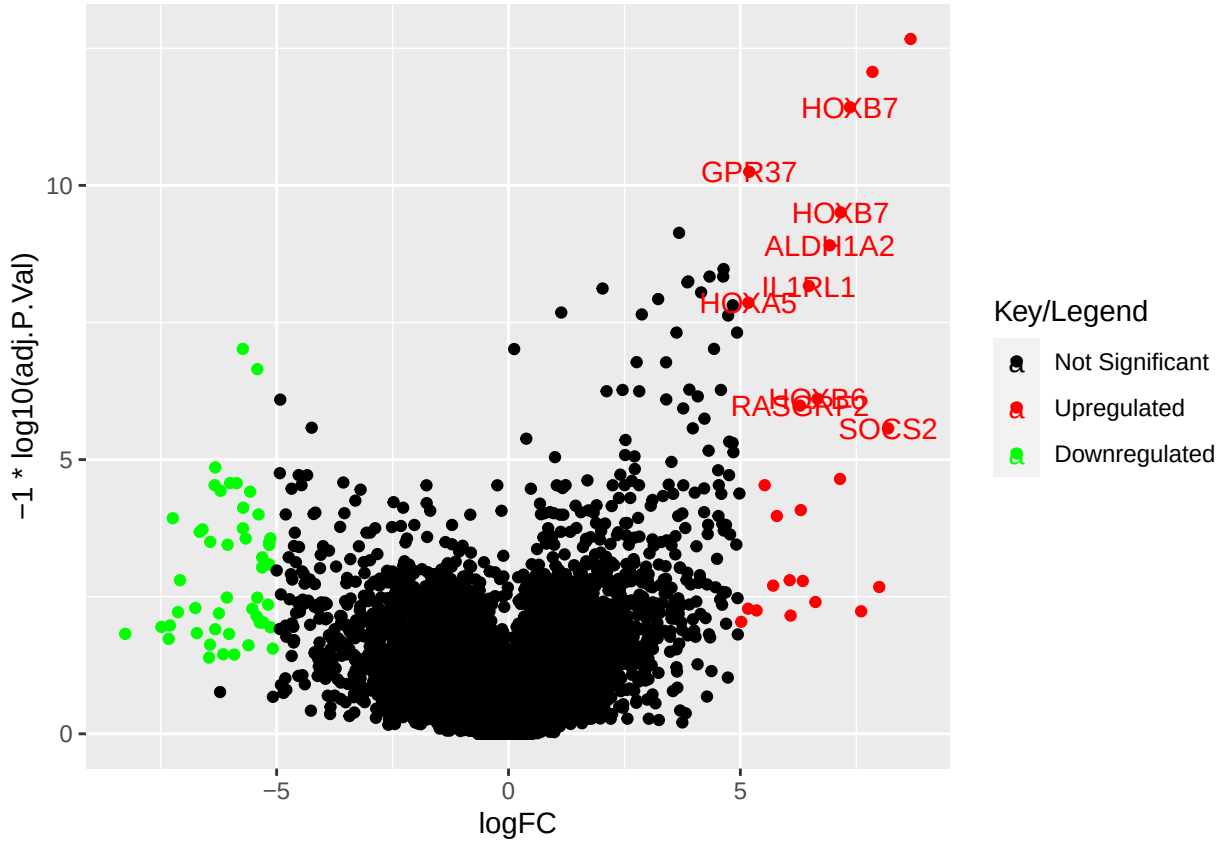
```
##
##     1     2     3
## 54601    25    49
```

```
volcano <- ggplot(data = results,
                  aes(x = logFC, y = -1*log10(adj.P.Val),
                      colour = threshold,
                      label = getsymbols))

volcano <- volcano +
  geom_point() +
  scale_color_manual(values = c("black", "red", "green"),
                     labels = c("Not Significant", "Upregulated", "Downregulated"),
                     name = "Key/Legend")

volcano +
  geom_text(data = subset(results, logFC > 5 & -1*log10(adj.P.Val) > 5), aes(x = logFC, y = -1*log10(adj
```

```
## Warning: Removed 2 rows containing missing values (`geom_text()`).
```


HOXB7
GPR37
HOXB7
ALDH1A2
IL1RL1
HOXA5
HOXB6
RASGRP2
SOCS2
10
5
0
−5
0
5
−1 * log10(adj.P.Val)
logFC
Key/Legend
Not Significant
Upregulated
Downregulated

4 Question 4

Try to find more information on the genes that are reported to be significantly differentially expressed. The place to start off is https://www.ncbi.nlm.nih.gov/gene/, remember that the data is from the species human. Try to look also for other databases where (some) information on the genes may be found. Try to follow on some of the provided links. Report in your own words on what you find. Report all the Gene Ontology (GO) terms associated with each gene. Are any of the GO terms common between genes? If so do the common GO terms seem to be related to anything particular? Try to present GO analysis in an informative manner, if possible visualize.

HOXB7:

- Offcial full name: homeobox B7
- Summary: "This gene is a member of the Antp homeobox family and encodes a protein with a homeobox DNA-binding domain. It is included in a cluster of homeobox B genes located on chromosome 17. The encoded nuclear protein functions as a sequence-specific transcription factor that is involved in cell proliferation and differentiation. Increased expression of this gene is associated with some cases of melanoma and ovarian carcinoma." (https://www.ncbi.nlm.nih.gov/gene/3217)
- GO terms (https://www.ncbi.nlm.nih.gov/gene/3217#gene-ontology):

Gene Ontology Provided by GOA

Function	Evidence Code
enables G protein-coupled peptide receptor activity	IBA
enables G protein-coupled peptide receptor activity	IDA
enables G protein-coupled receptor activity	TAS
enables Hsp70 protein binding	IPI
enables PDZ domain binding	IEA
enables heat shock protein binding	IPI
enables neuropeptide binding	IPI
enables neuropeptide receptor activity	IDA
enables peptide binding	IPI
enables prosaposin receptor activity	IBA
enables prosaposin receptor activity	IDA
enables protein binding	IPI
enables ubiquitin protein ligase binding	IPI

Figure 1: HOXB7

GPR37:

- Offcial full name: G protein-coupled receptor 37
- Summary: "This gene is a member of the G protein-coupled receptor family. The encoded protein contains seven transmembrane domains and is found in cell and endoplasmic reticulum membranes. G protein-coupled receptors are involved in translating outside signals into G protein mediated intracellular effects. This gene product interacts with Parkin and is involved in juvenile Parkinson disease." (https://www.ncbi.nlm.nih.gov/gene/2861)
- GO terms (https://www.ncbi.nlm.nih.gov/gene/2861#gene-ontology):

Gene Ontology Provided by GOA

Function	Evidence Code
enables DNA-binding transcription activator activity, RNA polymerase II-specific	IDA
enables DNA-binding transcription factor activity	NAS
enables DNA-binding transcription factor activity, RNA polymerase II-specific	IBA
enables DNA-binding transcription factor activity, RNA polymerase II-specific	ISA
enables RNA polymerase II cis-regulatory region sequence-specific DNA binding	IBA
enables RNA polymerase II cis-regulatory region sequence-specific DNA binding	IDA
enables protein binding	IPI
enables sequence-specific double-stranded DNA binding	IDA

Figure 2: GPR37

ALDH1A2:

- Offcial full name: aldehyde dehydrogenase 1 family member A2
- Summary: "This protein belongs to the aldehyde dehydrogenase family of proteins. The product of this gene is an enzyme that catalyzes the synthesis of retinoic acid (RA) from retinaldehyde. Retinoic acid, the active derivative of vitamin A (retinol), is a hormonal signaling molecule that functions in developing and adult tissues. The studies of a similar mouse gene suggest that this enzyme and the cytochrome CYP26A1, concurrently establish local embryonic retinoic acid levels which facilitate posterior organ development and prevent spina bifida. Four transcript variants encoding distinct isoforms have been identified for this gene." (https://www.ncbi.nlm.nih.gov/gene/8854)
- GO terms (https://www.ncbi.nlm.nih.gov/gene/8854#gene-ontology):

Gene Ontology Provided by GOA

Function	Evidence Code
enables 3-chloroallyl aldehyde dehydrogenase activity	ISS
enables aldehyde dehydrogenase (NAD+) activity	IBA
enables retinal binding	ISS
enables retinal dehydrogenase activity	IDA
enables retinal dehydrogenase activity	ISS

Figure 3: ALDH1A2

IL1RL1:

- Offcial full name: interleukin 1 receptor like 1
- Summary: "The protein encoded by this gene is a member of the interleukin 1 receptor family. Studies of the similar gene in mouse suggested that this receptor can be induced by proinflammatory stimuli, and may be involved in the function of helper T cells. This gene, interleukin 1 receptor, type I (IL1R1), interleukin 1 receptor, type II (IL1R2) and interleukin 1 receptor-like 2 (IL1RL2) form a cytokine receptor gene cluster in a region mapped to chromosome 2q12. Alternative splicing of this gene results in multiple transcript variants." (https://www.ncbi.nlm.nih.gov/gene/9173)
- GO terms (https://www.ncbi.nlm.nih.gov/gene/9173#gene-ontology):

Gene Ontology Provided by GOA

Function	Evidence Code
enables NAD+ nucleosidase activity	IEA
enables NAD+ nucleotidase, cyclic ADP-ribose generating	IEA
enables cytokine receptor activity	TAS
enables interleukin-1 receptor activity	IEA
enables interleukin-33 binding	IBA
enables interleukin-33 receptor activity	IDA
enables protein binding	IPI

Figure 4: IL1RL1

The common Gene Ontology (GO) term "Enables protein binding" appears across the following genes: HOXB7, GPR37 and IL1RL1.

Protein Binding is a important molecular function that allows proteins to interact with other molecules such as enzymes, receptor proteins. This GO term being common across these genes suggests that all the involved proteins might play significant roles in cellular signaling, metabolism, and structural processes.