

Gene Expression Analysis Using R

Case Study: Interferon regulatory factor 6 (*IRF6*)

Contents

0.1	Today's Objectives	2
0.2	Cleft Lip and Palate 1/3	2
0.3	Cleft Lip and Palate 2/3	2
0.4	Cleft Lip and Palate 3/3	2
0.5	Question	4
0.6	Hint	4
0.7	Hypothesis	4
0.8	Why Microarray?	4
0.9	Original Paper	4
0.10	Experimental Design	4
0.11	Dataset	5
0.12	Loading	6
0.13	Checking	6
0.14	Exploring	6
0.15	Transforming	7
0.16	Multiple Plots 1/2	8
0.17	Multiple Plots 2/2	9
0.18	Boxplot	9
0.19	Clustering 1/2	10
0.20	Clustering 2/2	11
0.21	Comparing	11
0.22	Scatter	11
0.23	Differentially Expressed Genes (DEGs)	11
0.24	Biological Significance (fold-change) 1/2	12
0.25	Biological Significance (fold-change) 2/2	12
0.26	Statistical Significance (<i>p</i> -value) 1/3	13
0.27	<i>t</i> -test	13
0.28	<i>t</i> -test : Example 1	15
0.29	<i>t</i> -test : Example 2	16
0.30	Statistical Significance (<i>p</i> -value) 2/3	16
0.31	Statistical Significance (<i>p</i> -value) 3/3	17
0.32	Volcano : Statistical & Biological 1/3	17
0.33	Volcano : Statistical & Biological 2/3	18
0.34	Volcano : Statistical & Biological 3/3	19
0.35	Filtering for DEGs 1/3	19
0.36	Filtering for DEGs 2/3	19
0.37	Filtering for DEGs 3/3	20
0.38	Exercise	20
0.39	Heatmap 1/5	20
0.40	Heatmap 2/5	21
0.41	Heatmap 3/5	21
0.42	Heatmap 4/5	22
0.43	Heatmap 5/5	22
0.44	Enhanced Heatmap 1/4	23
0.45	Enhanced Heatmap 2/4	23
0.46	Enhanced Heatmap 3/4	24

0.47 Enhanced Heatmap 4/4	25
0.48 Annotation 1/3	26
0.49 Annotation 2/3	26
0.50 Annotation 3/3	36
0.51 Exercise	36
0.52 Homework	36

0.1 Today's Objectives

- Download and load microarray dataset into R
- Explore the dataset with basic visualizations
- Identify differentially expressed genes (DEGs)
- Generate annotation of the DEGs



0.2 Cleft Lip and Palate 1/3

Cleft lip and cleft palate (**CLP**) are splits in the upper lip, the roof of the mouth (palate) or both. They result when facial structures that are developing in an unborn baby do not close completely. CLP is one of the most common birth defects with a frequency of 1/700 live births.

0.3 Cleft Lip and Palate 2/3

Children with cleft lip with or without cleft palate face a variety of challenges, depending on the type and severity of the cleft.

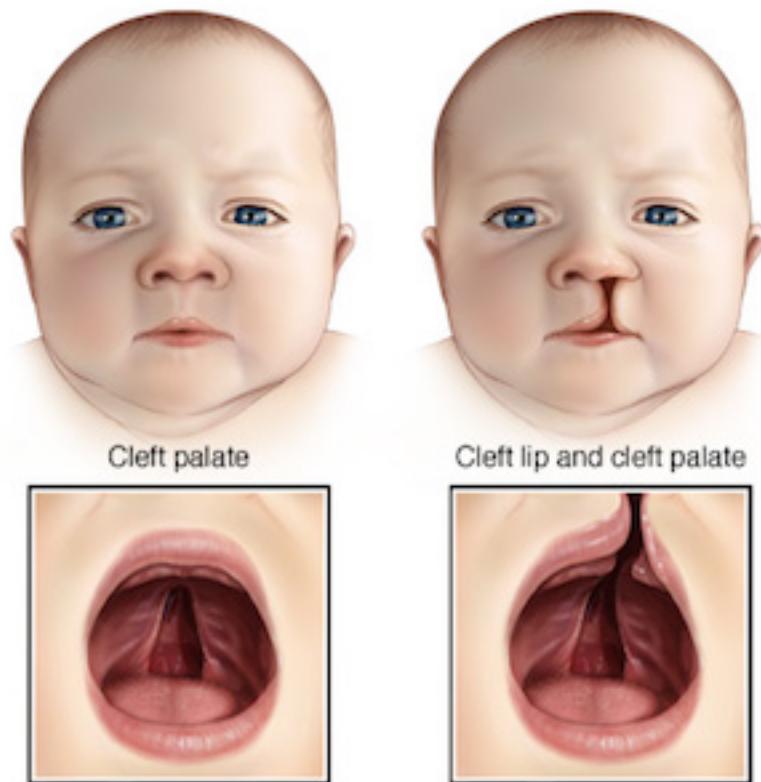
- **Difficulty feeding.** One of the most immediate concerns after birth is feeding.
- **Ear infections and hearing loss.** Babies with cleft palate are especially at risk of developing middle ear fluid and hearing loss.
- **Dental problems.** If the cleft extends through the upper gum, tooth development may be affected.
- **Speech difficulties.** Because the palate is used in forming sounds, the development of normal speech can be affected by a cleft palate. Speech may sound too nasal.

Reference: Mayo Foundation for Medical Education and Research

0.4 Cleft Lip and Palate 3/3

- DNA variation in Interferon Regulatory Factor 6 (**IRF6**) causes Van der Woude syndrome (**VWS**)
- VWS is the most common syndromic form of cleft lip and palate.
- However, the causing variant in IRF6 has been found in *only* 70% of VWS families!
- IRF6 is a **transcription factor** with a conserved helix-loop-helix DNA binding domain and a less well-conserved protein binding domain.

Reference: Hum Mol Genet. 2014 May 15; 23(10): 2711–2720



© MAYO FOUNDATION FOR MEDICAL EDUCATION AND RESEARCH. ALL RIGHTS RESERVED.

Figure 1: Cleft lip and palate

0.5 Question

Given:

1. The pathogenic variant in IRF6 exists in only 70% of the VWS families
2. IRF6 is a transcription factor

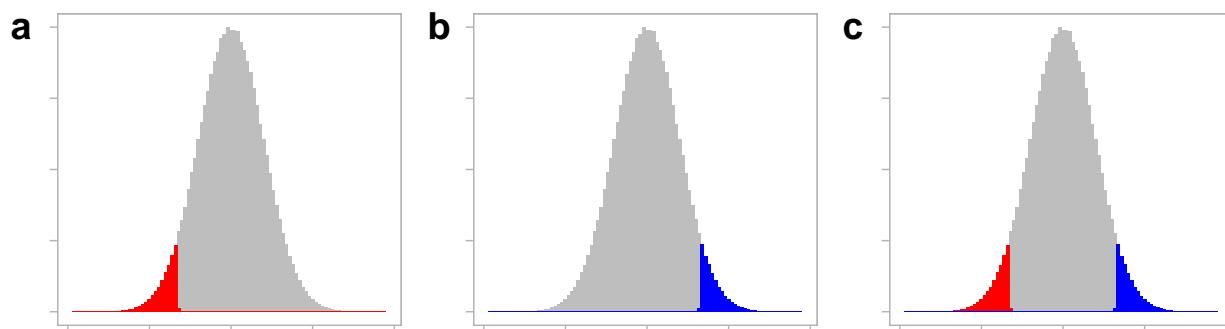
How can we identify other genes that might be involved in the remaining 30% of the VWS families?

0.6 Hint

- Usually, genes that are regulated by a transcription factor belong to the same biological process or pathway.
- Therefore, by comparing the gene expression patterns between wild-type (functional) *Irf6* and knockout (non-functional) *Irf6*, it could be possible to identify genes that are regulated (targeted) by *Irf6*.

0.7 Hypothesis

- $H_O : \mu_{WT} = \mu_{KO}$
- $H_A : \mu_{WT} \neq \mu_{KO}$
- Where μ is the *mean* of the gene expression values of a gene.
- **One-sided or Two-sided testing?**



0.8 Why Microarray?

- It does not require a predefined set of candidate genes
- It requires only a small amount of RNA
- It is a high-throughput protocol - transcriptome-wide analysis
- One experiment can generate multiple hypotheses

0.9 Original Paper

0.10 Experimental Design

- 3 IRF6 wild-type (+/+) and 3 knockout (-/-) mouse embryos.
- E17.5 embryos were removed from euthanized mothers.

Abnormal skin, limb and craniofacial morphogenesis in mice deficient for interferon regulatory factor 6 (*Irf6*).

Ingraham CR¹, Kinoshita A, Kondo S, Yang B, Sajan S, Trout KJ, Malik MI, Dunnwald M, Goudy SL, Lovett M, Murray JC, Schutte BC.

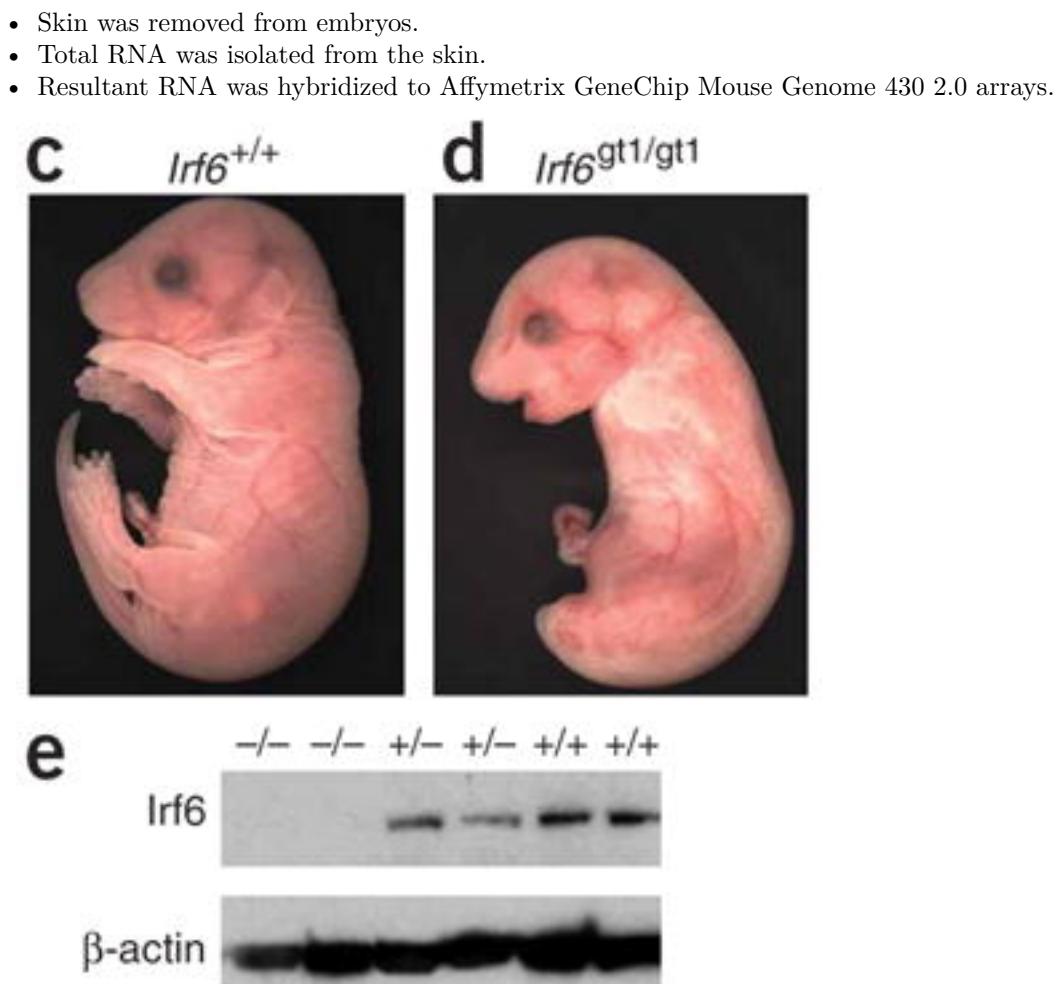
 Author information

Abstract

Transcription factor paralogs may share a common role in staged or overlapping expression in specific tissues, as in the Hox family. In other cases, family members have distinct roles in a range of embryologic, differentiation or response pathways (as in the Tbx and Pax families). For the interferon regulatory factor (IRF) family of transcription factors, mice deficient in *Irf1*, *Irf2*, *Irf3*, *Irf4*, *Irf5*, *Irf7*, *Irf8* or *Irf9* have defects in the immune response but show no embryologic abnormalities. Mice deficient for *Irf6* have not been reported, but in humans, mutations in *IRF6* cause two mendelian orofacial clefting syndromes, and genetic variation in *IRF6* confers risk for isolated cleft lip and palate. Here we report that mice deficient for *Irf6* have abnormal skin, limb and craniofacial development. Histological and gene expression analyses indicate that the primary defect is in keratinocyte differentiation and proliferation. This study describes a new role for an IRF family member in epidermal development.

PMID: 17041601 PMCID: PMC2082114 DOI: 10.1038/ng1903

Figure 2: PMID: 17041601



0.11 Dataset

- The original dataset can be obtained from NCBI GEO with accession GSE5800

ID	KO1	KO2	KO3	WT1	WT2	WT3
1415670_at	6531.0	5562.8	6822.4	7732.1	7191.2	7551.9
1415671_at	11486.3	10542.7	10641.4	10408.2	9484.5	7650.2
1415672_at	14339.2	13526.1	14444.7	12936.6	13841.7	13285.7
1415673_at	3156.8	2219.5	3264.4	2374.2	2201.8	2525.3

- Download the dataset from the following link <https://goo.gl/gH7QLM>

0.12 Loading

We are going to load the dataset from a tsv file (`Irf6.tsv`) into a variable called `data` using function `read.table`. `data` here is just an arbitrary **variable** name to hold the result of `read.table` and it can be called/named *almost* anything. See The State of Naming Conventions in R (Bååth 2012) for more information on naming **variables** in R.

```
# Load data from text file into a varilable
data = as.matrix(read.table("Irf6.tsv", header = TRUE, row.names = 1))
```

Note: the hash sign (#) indicates that what comes after is a *comment*. Comments are for documentation and readability of the R code and they are not evaluated (or executed).

0.13 Checking

```
dim(data) # Dimension of the dataset

## [1] 45101      6

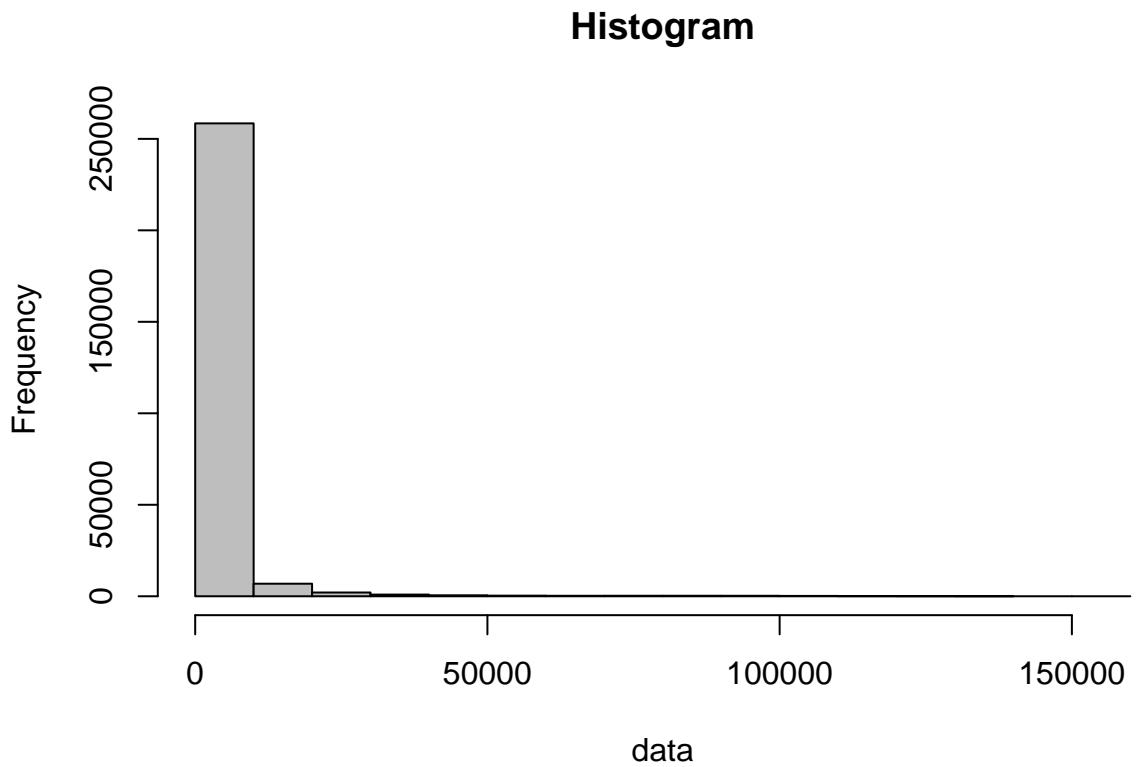
head(data) # First few rows
```

	KO1	KO2	KO3	WT1	WT2	WT3
1415670_at	6531.0	5562.8	6822.4	7732.1	7191.2	7551.9
1415671_at	11486.3	10542.7	10641.4	10408.2	9484.5	7650.2
1415672_at	14339.2	13526.1	14444.7	12936.6	13841.7	13285.7
1415673_at	3156.8	2219.5	3264.4	2374.2	2201.8	2525.3
1415674_a_at	4002.0	3306.9	3777.0	3760.6	3137.0	2911.5
1415675_at	3468.4	3347.4	3332.9	3073.5	3046.0	2914.4

0.14 Exploring

Check the behavior of the data (e.g., normal?, skewed?)

```
hist(data, col = "gray", main="Histogram")
```

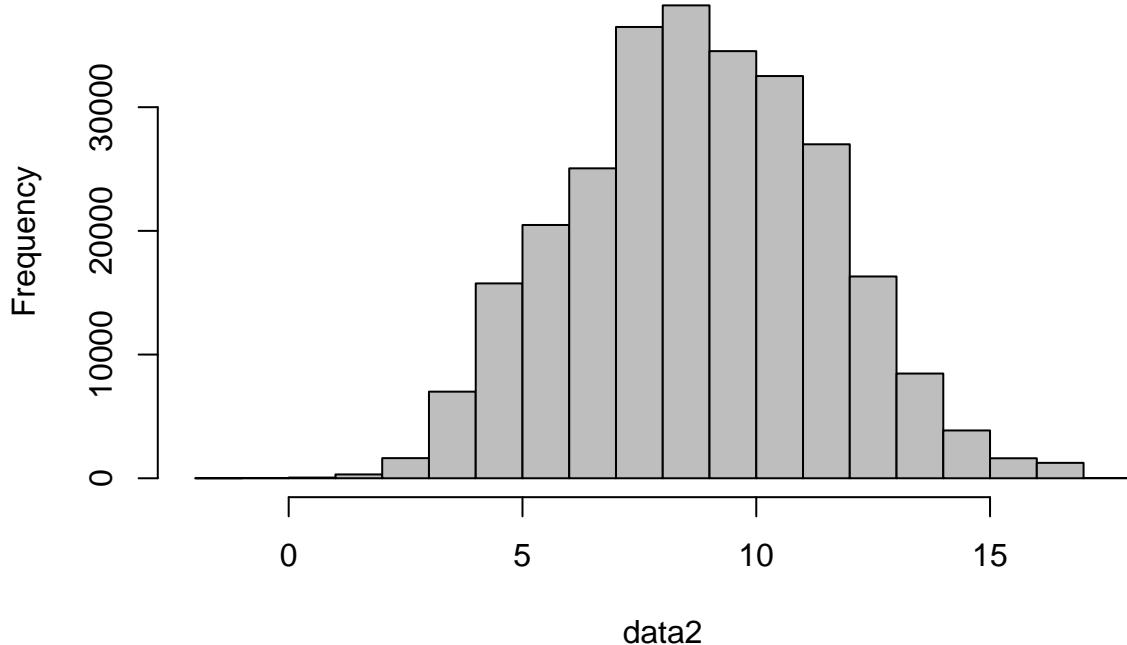


0.15 Transforming

\log_2 transformation (why?)

```
data2 = log2(data)
hist(data2, col = "gray", main="Histogram")
```

Histogram



0.16 Multiple Plots 1/2

```
samples = colnames(data2) # Headers (names) of the columns
samples

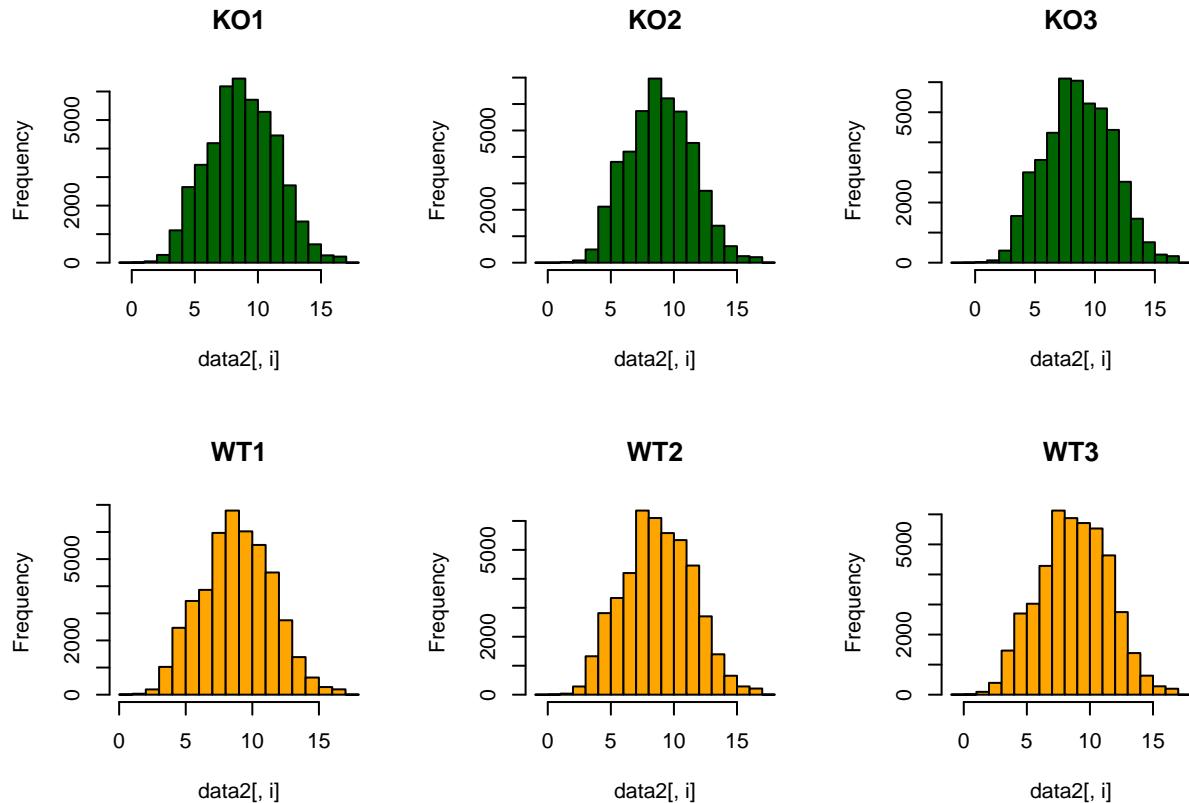
## [1] "K01" "K02" "K03" "WT1" "WT2" "WT3"

par( mfrow = c( 2, 3 ) ) # Split screen into 2 rows x 3 columns partitions

for (i in 1:3) {
  # for each of the first 3 columns in the table
  hist(data2[,i], col = "red", main = samples[i])
}

for (i in 4:6) {
  # for each of the last 3 columns in the table
  hist(data2[,i], col = "green", main = samples[i])
}
```

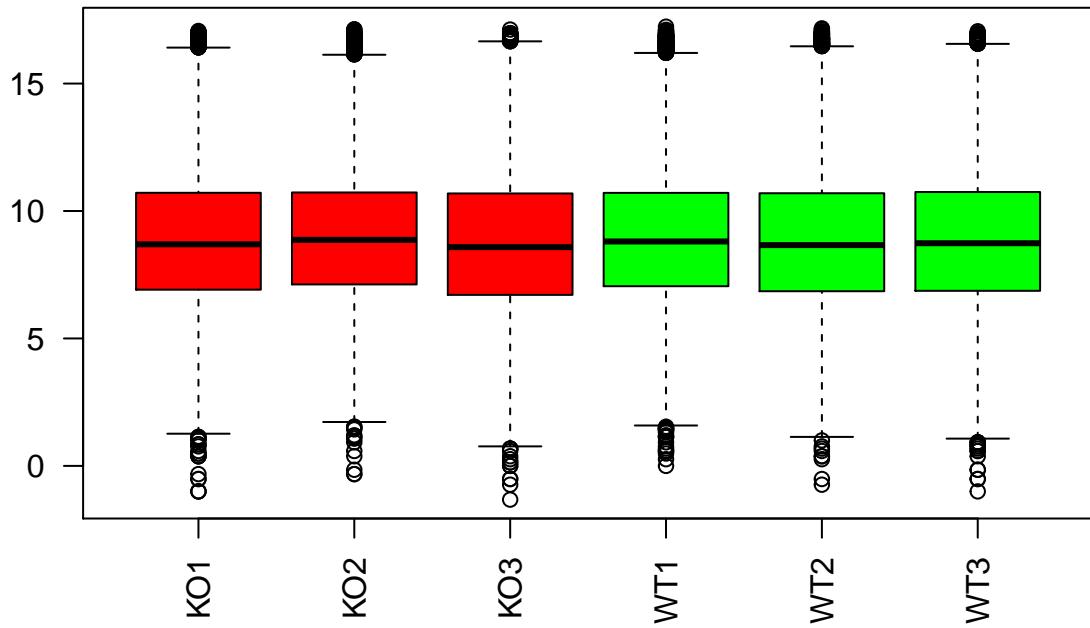
0.17 Multiple Plots 2/2



```
par( mfrow = c( 1, 1 ) ) # Just to set screen back to 1 partition
```

0.18 Boxplot

```
colors = c(rep("red", 3), rep("green", 3))
boxplot(data2, col = colors, las = 2)
```

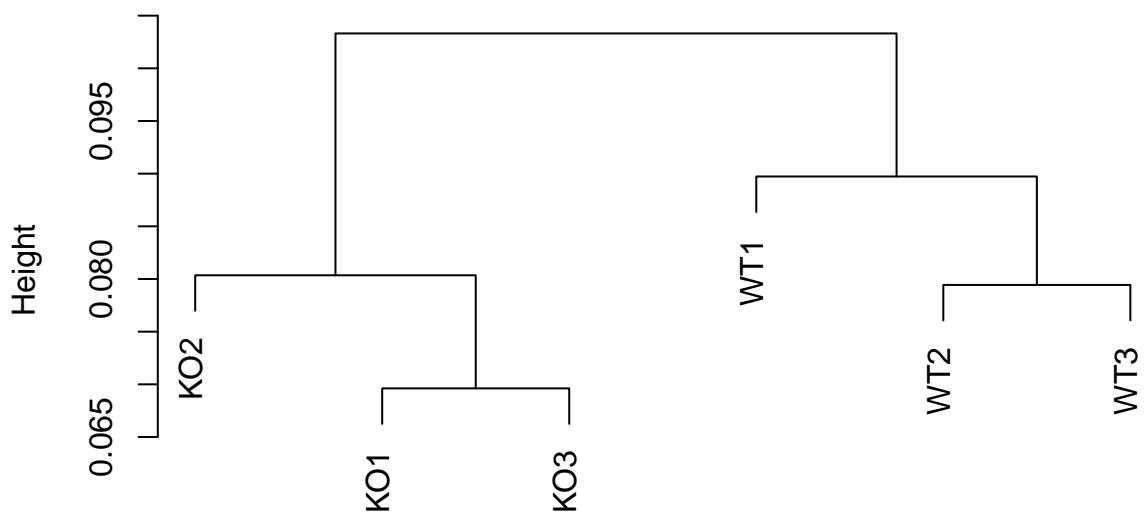


0.19 Clustering 1/2

Hierarchical clustering of the samples (i.e., columns) based on the correlation coefficients of the expression values

```
hc = hclust(as.dist(1 - cor(data2)))
plot(hc)
```

Cluster Dendrogram



```
as.dist(1 - cor(data2))
hclust (*, "complete")
```

0.20 Clustering 2/2

To learn about a function (e.g., `hclust`), you may type `?function` (e.g., `?hclust`) in the console to launch R documentation on that function:

0.21 Comparing

We are going to compare the **means** of the replicates of the two conditions

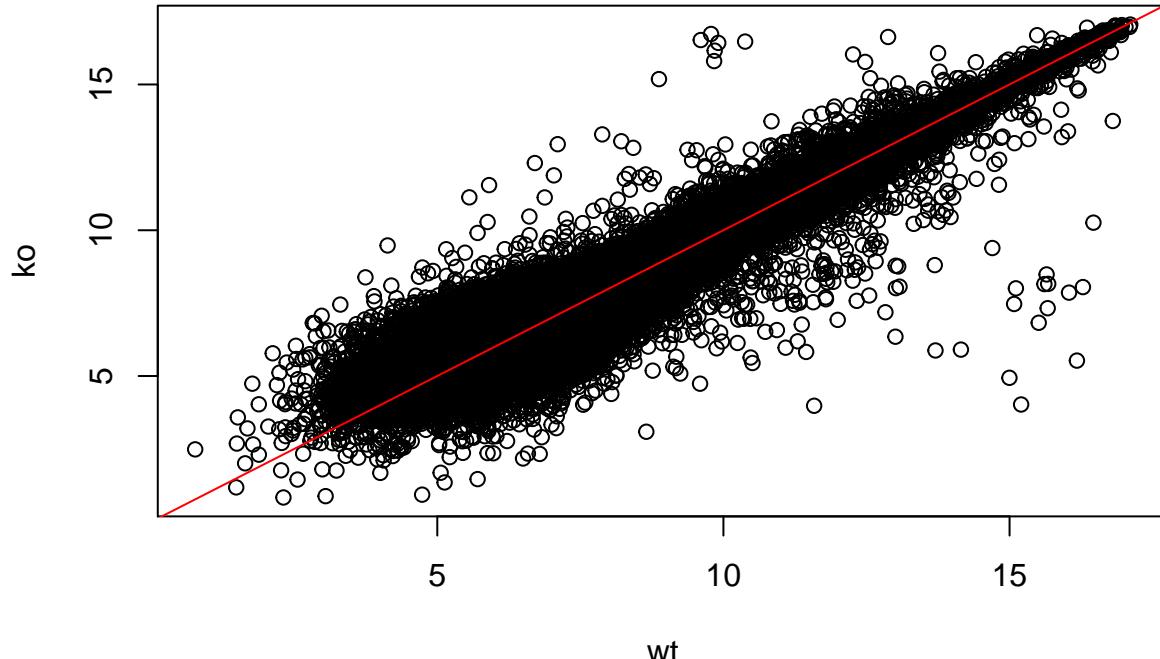
```
# Compute the means of the samples of each condition
ko = apply(data2[, 1:3], 1, mean)
head(ko)

##    1415670_at    1415671_at    1415672_at    1415673_at 1415674_a_at
##    12.61692      13.40966     13.78313     11.47096     11.84693
##    1415675_at
##    11.72381

wt = apply(data2[, 4:6], 1, mean)
head(wt)

##    1415670_at    1415671_at    1415672_at    1415673_at 1415674_a_at
##    12.87043      13.15269     13.70450     11.20664     11.66649
##    1415675_at
##    11.55578
```

0.22 Scatter



0.23 Differentially Expressed Genes (DEGs)

To identify DEGs, we will identify:

- **Biologically** significantly differentially expressed
- **Statistically** significantly differentially expressed

Then, we will take the **overlap (intersection)** of the two sets

0.24 Biological Significance (fold-change) 1/2

```
fold = ko - wt # Difference between means
head(fold)

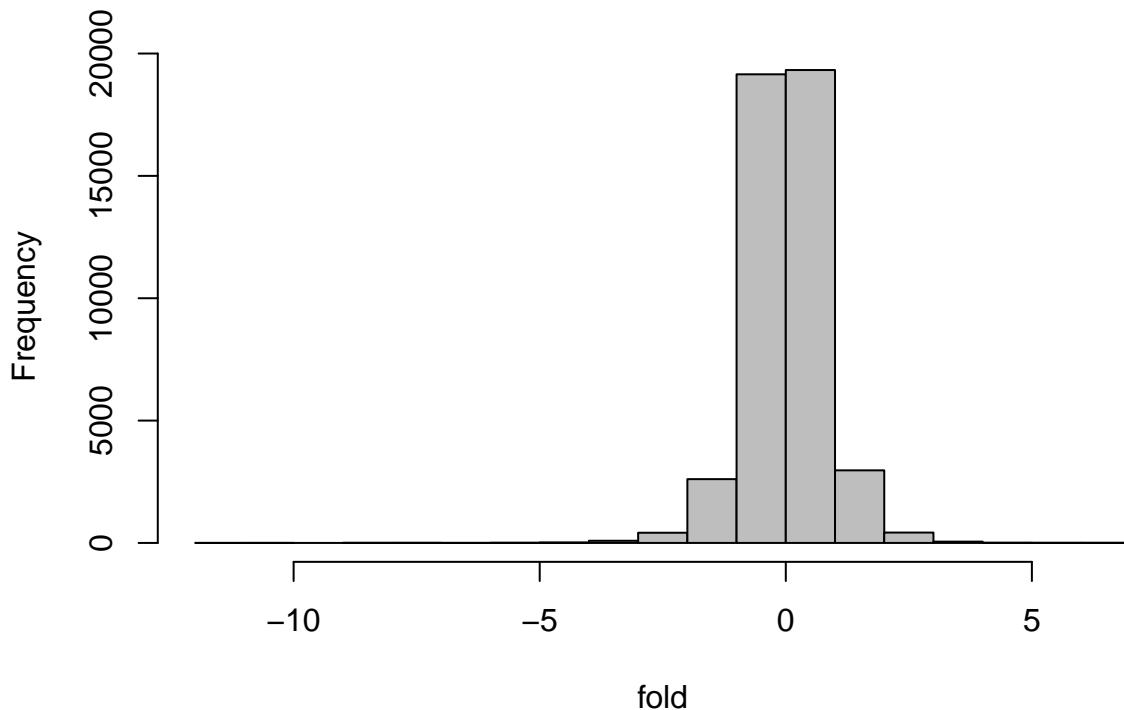
##    1415670_at    1415671_at    1415672_at    1415673_at 1415674_a_at
##   -0.25351267    0.25697097    0.07863227    0.26431191   0.18044345
##    1415675_at
##   0.16803065
```

- What do the positive and negative values of the fold-change indicate? Considering the WT condition is the **reference (or control)**
 - +ve → Up-regulation ↑
 - -ve → Down-regulation ↓

0.25 Biological Significance (fold-change) 2/2

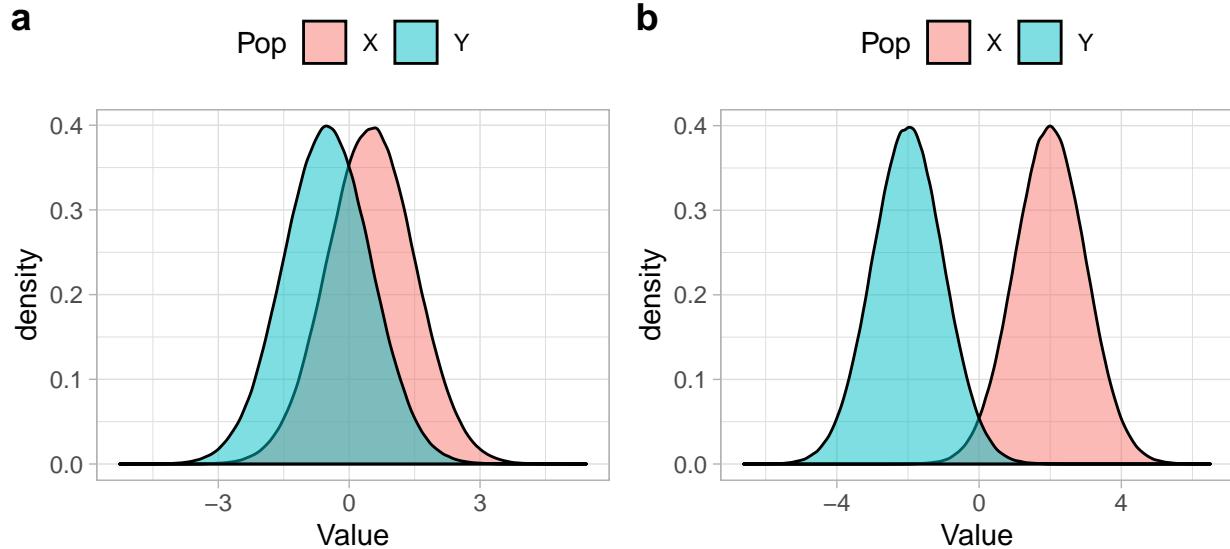
```
hist(fold, col = "gray") # Histogram of the fold
```

Histogram of fold



0.26 Statistical Significance (p -value) 1/3

- To assess the statistical significance of the difference in the expression values for each gene between the two conditions (e.g., WT and K0), we are going to use t -test.



0.27 t -test

Let's say there are two samples x and y from the two populations, X and Y , respectively, to determine whether the means of two populations are significantly different, we can use `t.test`.

```
?t.test

## Student's t-Test
##
## Description:
##
##     Performs one and two sample t-tests on vectors of data.
##
## Usage:
##
##     t.test(x, ...)
##
##     ## Default S3 method:
##     t.test(x, y = NULL,
##             alternative = c("two.sided", "less", "greater"),
##             mu = 0, paired = FALSE, var.equal = FALSE,
##             conf.level = 0.95, ...)
##
##     ## S3 method for class 'formula'
##     t.test(formula, data, subset, na.action, ...)
##
## Arguments:
##
##     x: a (non-empty) numeric vector of data values.
##
##     y: an optional (non-empty) numeric vector of data values.
```

```

##
## alternative: a character string specifying the alternative hypothesis,
##               must be one of '"two.sided"' (default), '"greater"' or
##               '"less"'. You can specify just the initial letter.
##
## mu: a number indicating the true value of the mean (or difference
##      in means if you are performing a two sample test).
##
## paired: a logical indicating whether you want a paired t-test.
##
## var.equal: a logical variable indicating whether to treat the two
##            variances as being equal. If 'TRUE' then the pooled variance
##            is used to estimate the variance otherwise the Welch (or
##            Satterthwaite) approximation to the degrees of freedom is
##            used.
##
## conf.level: confidence level of the interval.
##
## formula: a formula of the form 'lhs ~ rhs' where 'lhs' is a numeric
##           variable giving the data values and 'rhs' a factor with two
##           levels giving the corresponding groups.
##
## data: an optional matrix or data frame (or similar: see
##       'model.frame') containing the variables in the formula
##       'formula'. By default the variables are taken from
##       'environment(formula)'.
##
## subset: an optional vector specifying a subset of observations to be
##         used.
##
## na.action: a function which indicates what should happen when the data
##            contain 'NA's. Defaults to 'getOption("na.action")'.
##
##       ....: further arguments to be passed to or from methods.
##
## Details:
##
##       The formula interface is only applicable for the 2-sample tests.
##
##       'alternative = "greater"' is the alternative that 'x' has a larger
##       mean than 'y'.
##
##       If 'paired' is 'TRUE' then both 'x' and 'y' must be specified and
##       they must be the same length. Missing values are silently removed
##       (in pairs if 'paired' is 'TRUE'). If 'var.equal' is 'TRUE' then
##       the pooled estimate of the variance is used. By default, if
##       'var.equal' is 'FALSE' then the variance is estimated separately
##       for both groups and the Welch modification to the degrees of
##       freedom is used.
##
##       If the input data are effectively constant (compared to the larger
##       of the two means) an error is generated.
##
## Value:

```

```

##
##      A list with class '"htest"' containing the following components:
##
## statistic: the value of the t-statistic.
##
## parameter: the degrees of freedom for the t-statistic.
##
## p.value: the p-value for the test.
##
## conf.int: a confidence interval for the mean appropriate to the
##           specified alternative hypothesis.
##
## estimate: the estimated mean or difference in means depending on
##           whether it was a one-sample test or a two-sample test.
##
## null.value: the specified hypothesized value of the mean or mean
##           difference depending on whether it was a one-sample test or a
##           two-sample test.
##
## stderr: the standard error of the mean (difference), used as
##           denominator in the t-statistic formula.
##
## alternative: a character string describing the alternative hypothesis.
##
## method: a character string indicating what type of t-test was
##           performed.
##
## data.name: a character string giving the name(s) of the data.
##
## See Also:
##
##      'prop.test'
##
## Examples:
##
##      require(graphics)
##
##      t.test(1:10, y = c(7:20))      # P = .00001855
##      t.test(1:10, y = c(7:20, 200)) # P = .1245      -- NOT significant anymore
##
##      ## Classical example: Student's sleep data
##      plot(extra ~ group, data = sleep)
##      ## Traditional interface
##      with(sleep, t.test(extra[group == 1], extra[group == 2]))
##      ## Formula interface
##      t.test(extra ~ group, data = sleep)

```

0.28 *t*-test : Example 1

```
x = c(4, 3, 10, 7, 9) ; y = c(7, 4, 3, 8, 10)
t.test(x, y)
```

```
##
```

```

## Welch Two Sample t-test
##
## data: x and y
## t = 0.1066, df = 7.9743, p-value = 0.9177
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -4.12888 4.52888
## sample estimates:
## mean of x mean of y
##       6.6        6.4
t.test(x, y)$p.value

## [1] 0.917739

```

0.29 *t*-test : Example 2

```

x = c(6, 8, 10, 7, 9) ; y = c(3, 2, 1, 4, 5)
t.test(x, y)

##
## Welch Two Sample t-test
##
## data: x and y
## t = 5, df = 8, p-value = 0.001053
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 2.693996 7.306004
## sample estimates:
## mean of x mean of y
##       8        3
t.test(x, y)$p.value

## [1] 0.001052826

```

0.30 Statistical Significance (*p*-value) 2/3

```

pvalue = NULL # Empty list for the p-values
n = nrow(data) # Number of genes (rows)
n

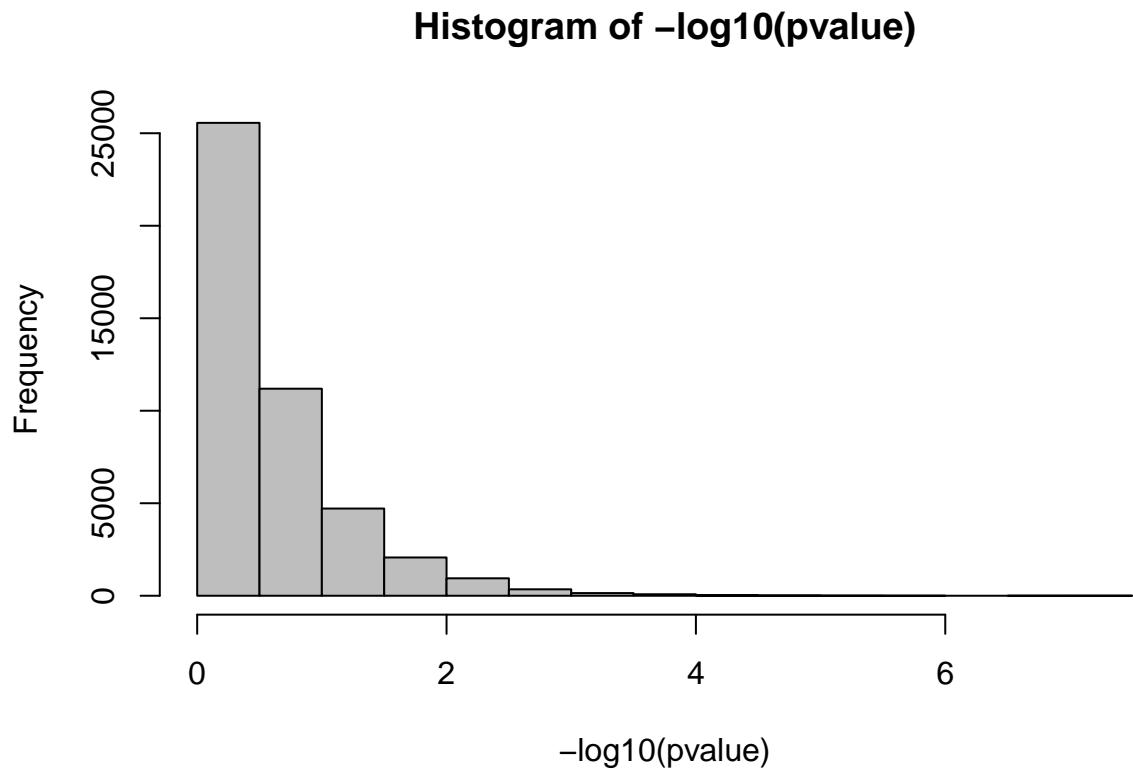
## [1] 45101
for(i in 1 : n) {
  # for each gene
  x = data2[i, 1:3] # wt values of gene number i
  y = data2[i, 4:6] # ko values of gene number i
  t = t.test(x, y) # t-test between the two conditions
  pvalue[i] = t$p.value # Put current p-value in the list of p-values
}
head(pvalue)

## [1] 0.092706280 0.182663337 0.129779075 0.272899180 0.262377176 0.005947807

```

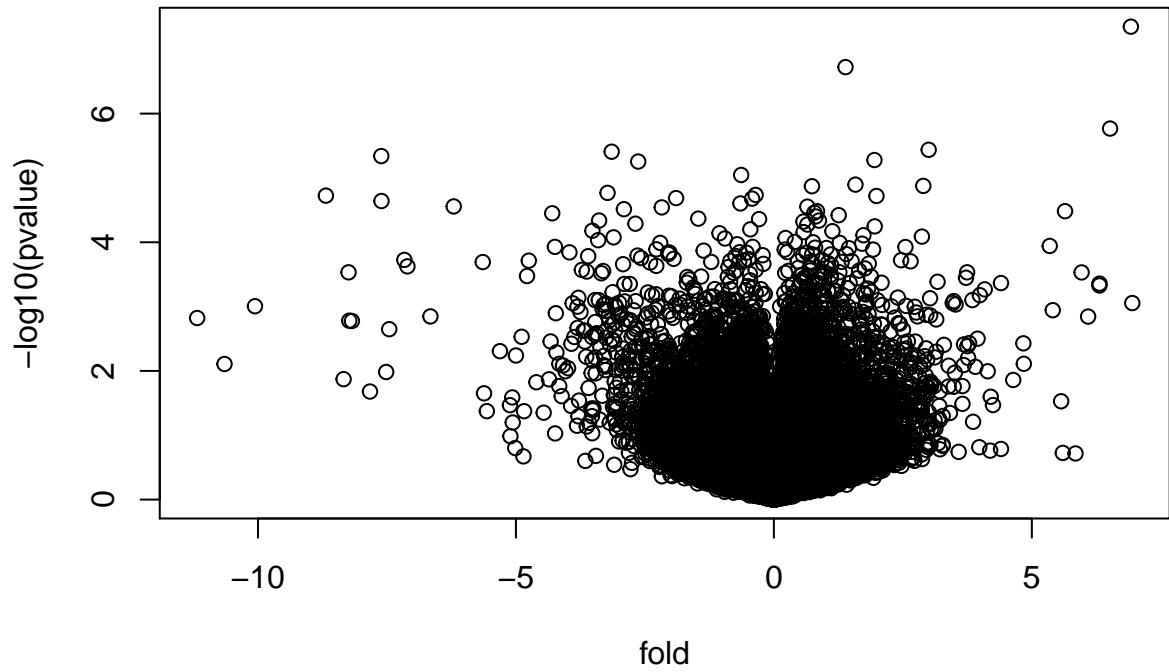
0.31 Statistical Significance (p -value) 3/3

```
hist(-log10(pvalue), col = "gray") # Histogram of p-values (-log10)
```



0.32 Volcano : Statistical & Biological 1/3

```
plot(-log10(pvalue) ~ fold)
```



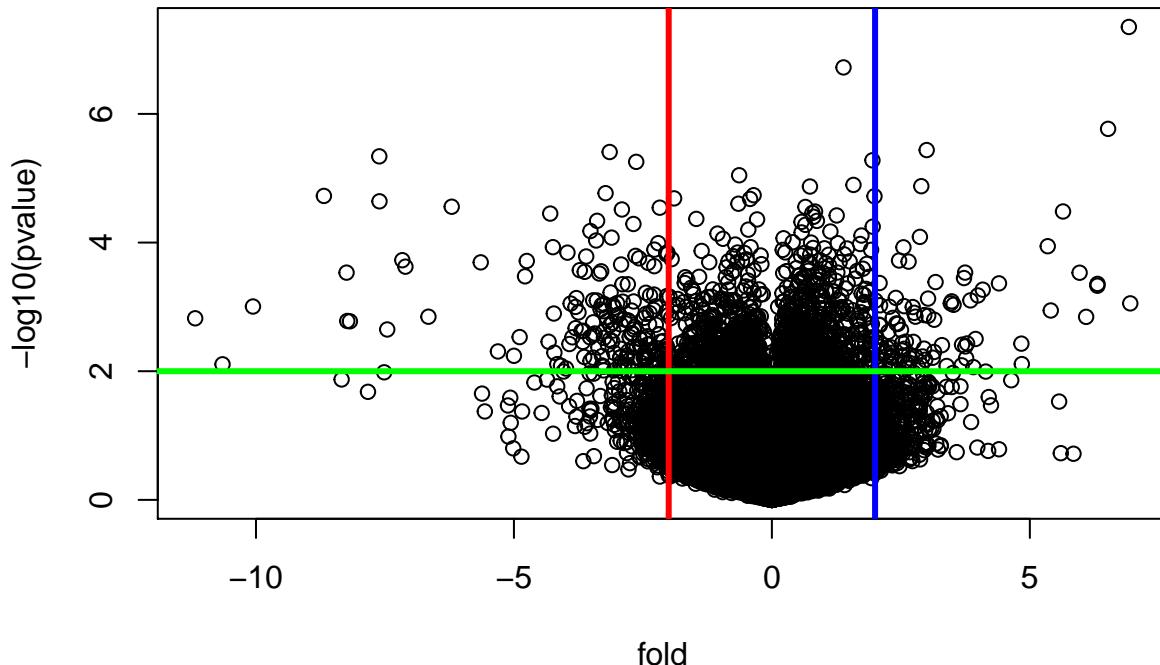
0.33 Volcano : Statistical & Biological 2/3

```
fold_cutoff = 2
pvalue_cutoff = 0.01

plot(-log10(pvalue) ~ fold)

abline(v = fold_cutoff, col = "blue", lwd = 3)
abline(v = -fold_cutoff, col = "red", lwd = 3)
abline(h = -log10(pvalue_cutoff), col = "green", lwd = 3)
```

0.34 Volcano : Statistical & Biological 3/3



0.35 Filtering for DEGs 1/3

```
filter_by_fold = abs(fold) >= fold_cutoff # Biological
sum(filter_by_fold) # Number of genes satisfy the condition

## [1] 1051

filter_by_pvalue = pvalue <= pvalue_cutoff # Statistical
sum(filter_by_pvalue)

## [1] 1564

filter_combined = filter_by_fold & filter_by_pvalue # Combined
sum(filter_combined)

## [1] 276
```

0.36 Filtering for DEGs 2/3

```
filtered = data2[filter_combined, ]
dim(filtered)

## [1] 276   6

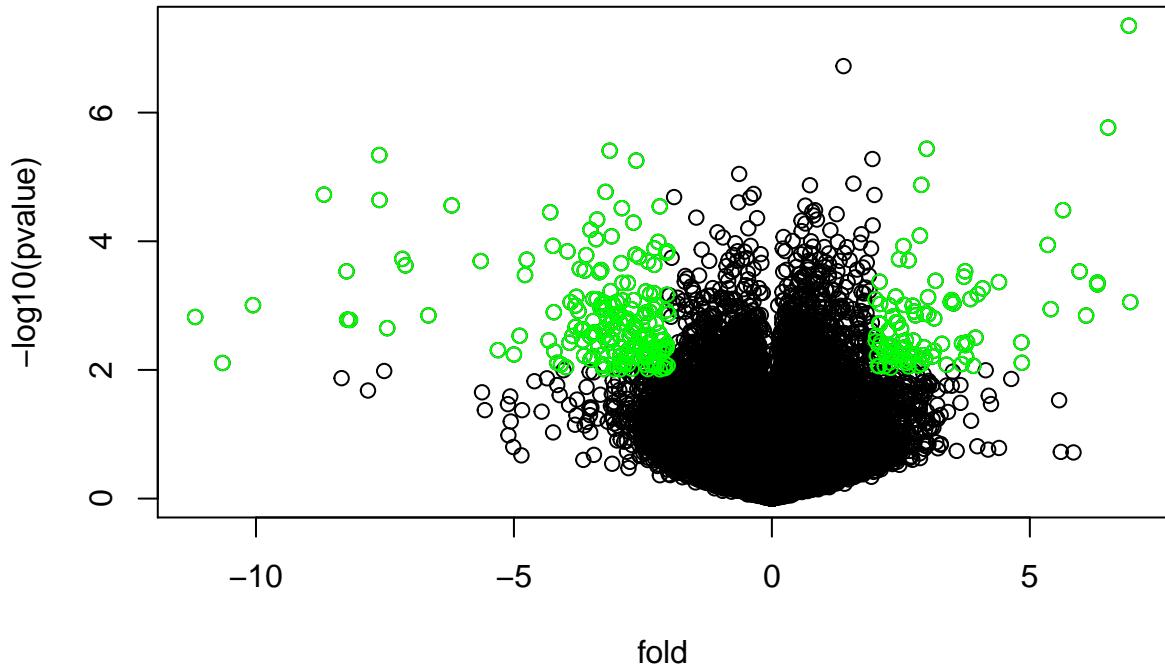
head(filtered)
```

	KO1	KO2	KO3	WT1	WT2	WT3
1416200_at	13.312004	12.973357	12.868456	7.40429	8.558803	8.683696
1416236_a_at	14.148397	14.039236	14.130007	12.23604	12.022403	11.495055

	KO1	KO2	KO3	WT1	WT2	WT3
1417808_at	5.321928	5.442944	4.053111	15.16978	15.070087	14.753274
1417932_at	10.602884	10.257152	10.496055	13.98445	14.203295	13.720960
1418050_at	10.622052	10.975490	10.795066	12.86513	13.012048	12.658122
1418100_at	9.117903	8.634811	9.057721	12.90358	12.842449	12.233769

0.37 Filtering for DEGs 3/3

```
plot(-log10(pvalue) ~ fold)
points(-log10(pvalue[filter_combined]) ~ fold[filter_combined],
       col = "green")
```

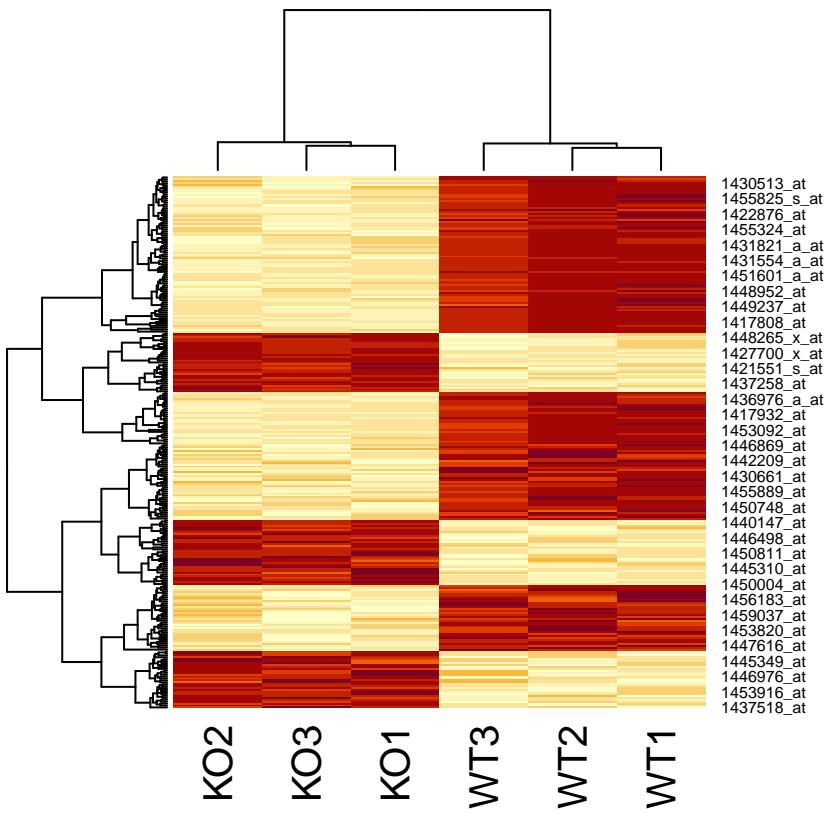


0.38 Exercise

On the volcano plot, highlight the up-regulated genes in red and the down-regulated genes in blue

0.39 Heatmap 1/5

```
heatmap(filtered)
```

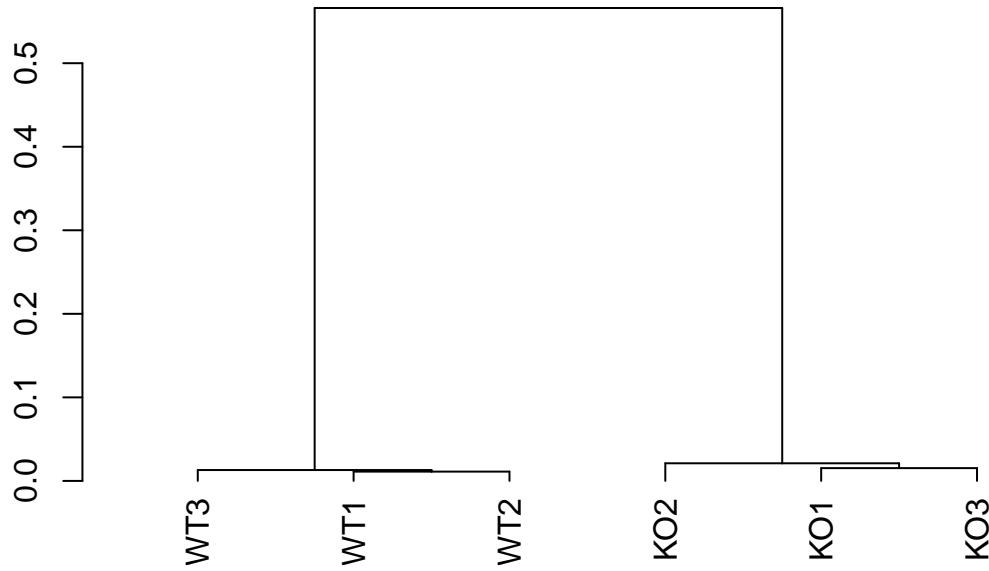


0.40 Heatmap 2/5

- By default, `heatmap` clusters genes (rows) and samples (columns) based on the Euclidean distance.
- In the context of gene expression, it is more appropriate to cluster the genes and samples based on the correlation to explore patterns of **co-regulation (co-expression)** among the genes and similarities between samples.
- To let `heatmap` cluster the genes and/or samples, we need to cluster the genes and samples by correlation coefficients (using `cor`) among the genes and samples.

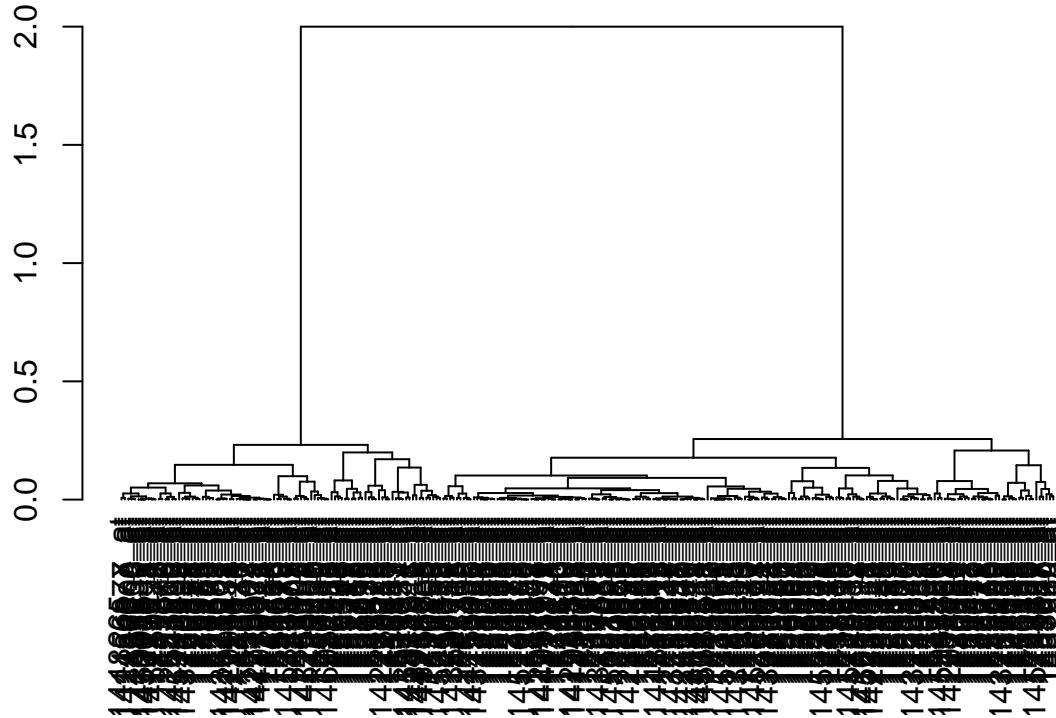
0.41 Heatmap 3/5

```
# Clustering of the columns (samples)
col_dendrogram = as.dendrogram(hclust(as.dist(1-cor(filtered))))
plot(col_dendrogram)
```



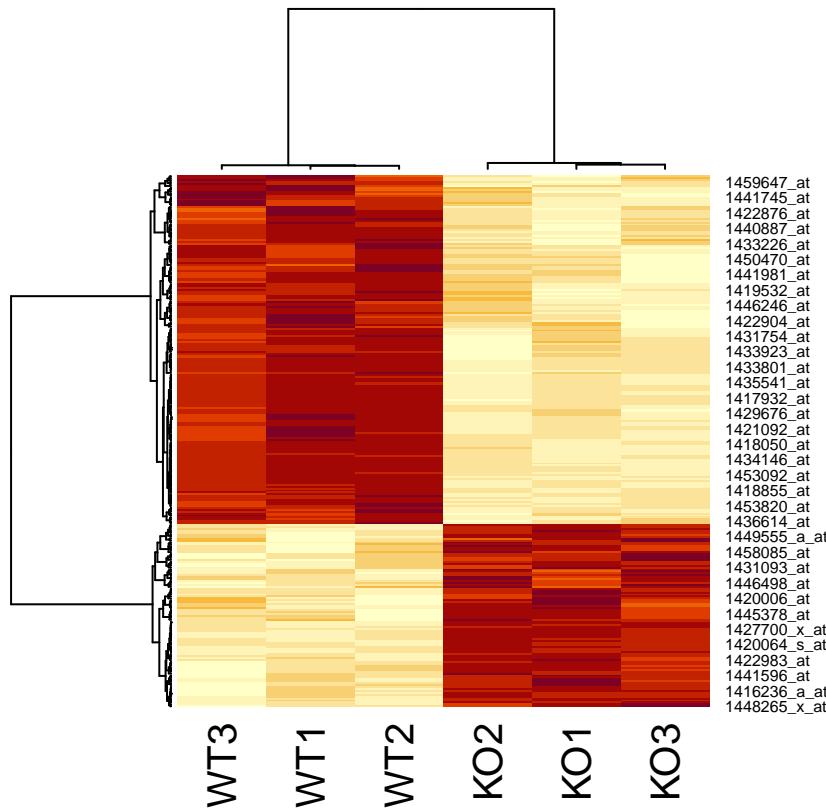
0.42 Heatmap 4/5

```
# Clustering of the rows (genes)
row_dendrogram = as.dendrogram(hclust(as.dist(1-cor(t(filtered)))))
plot(row_dendrogram)
```



0.43 Heatmap 5/5

```
# Heatmap with the rows and columns clustered by correlation coefficients
heatmap(filtered, Rowv=row_dendrogram, Colv=col_dendrogram)
```



0.44 Enhanced Heatmap 1/4

There is an *enhanced* heatmap function `heatmap.2` that comes with the `gplots` package. `heatmap.2` provides more options via a richer set of parameters.

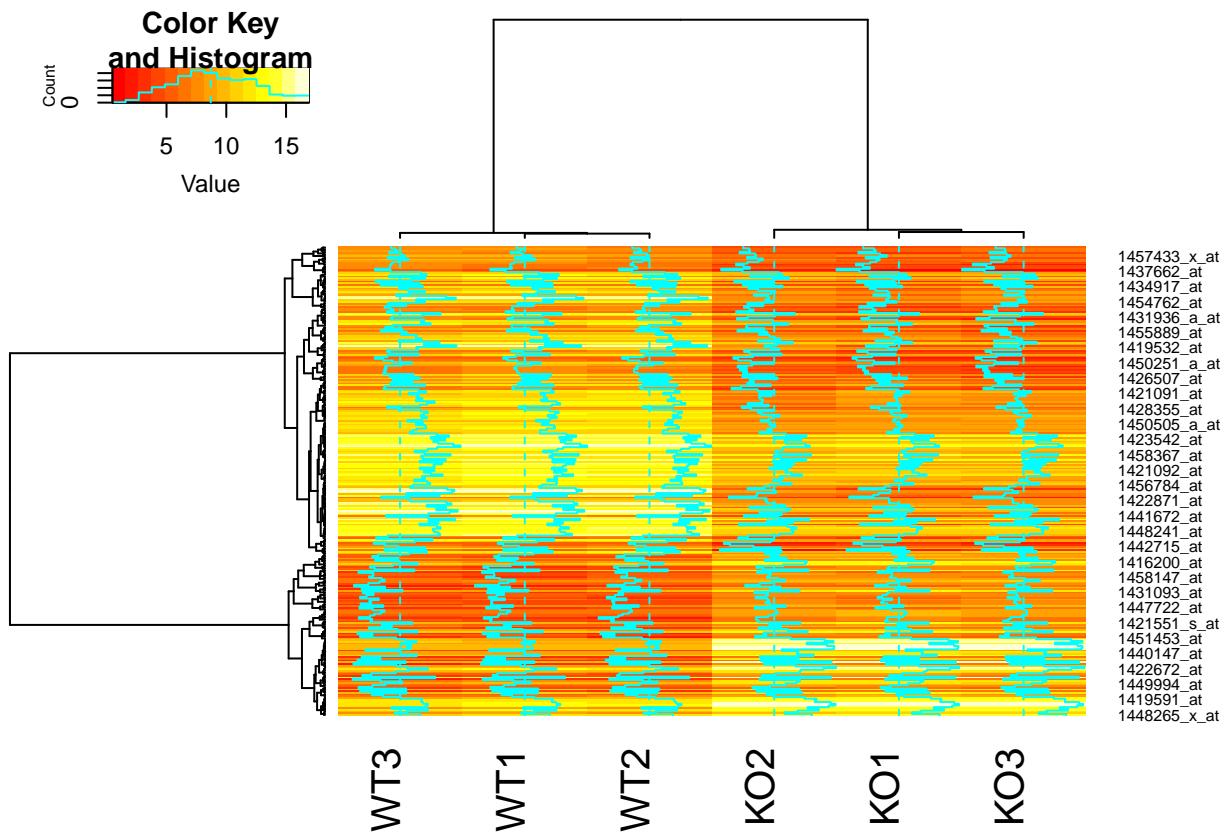
```
install.packages("gplots") # Install the package (if not already installed)
```

```
library(gplots) # Load the package
```

```
##
## Attaching package: 'gplots'
## The following object is masked from 'package:stats':
##      lowess
```

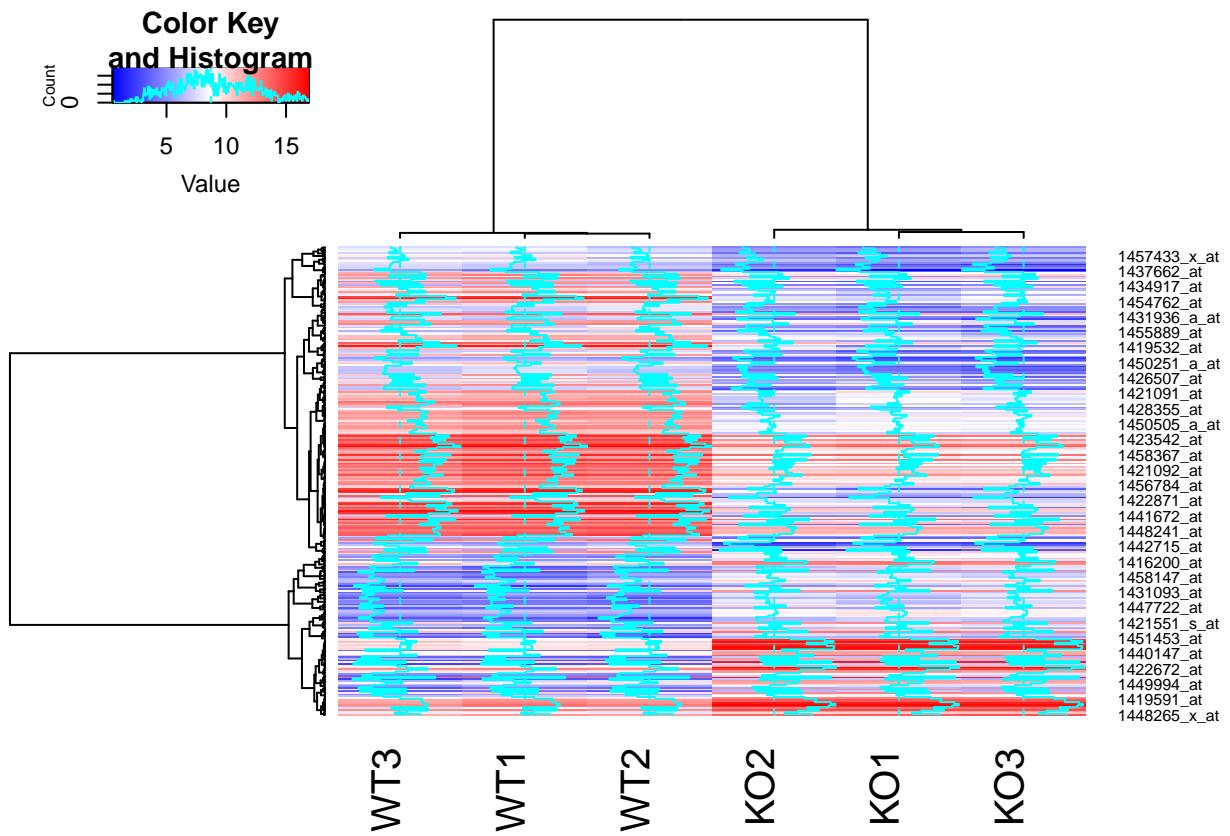
0.45 Enhanced Heatmap 2/4

```
heatmap.2(filtered, Rowv=row_dendrogram, Colv=col_dendrogram)
```



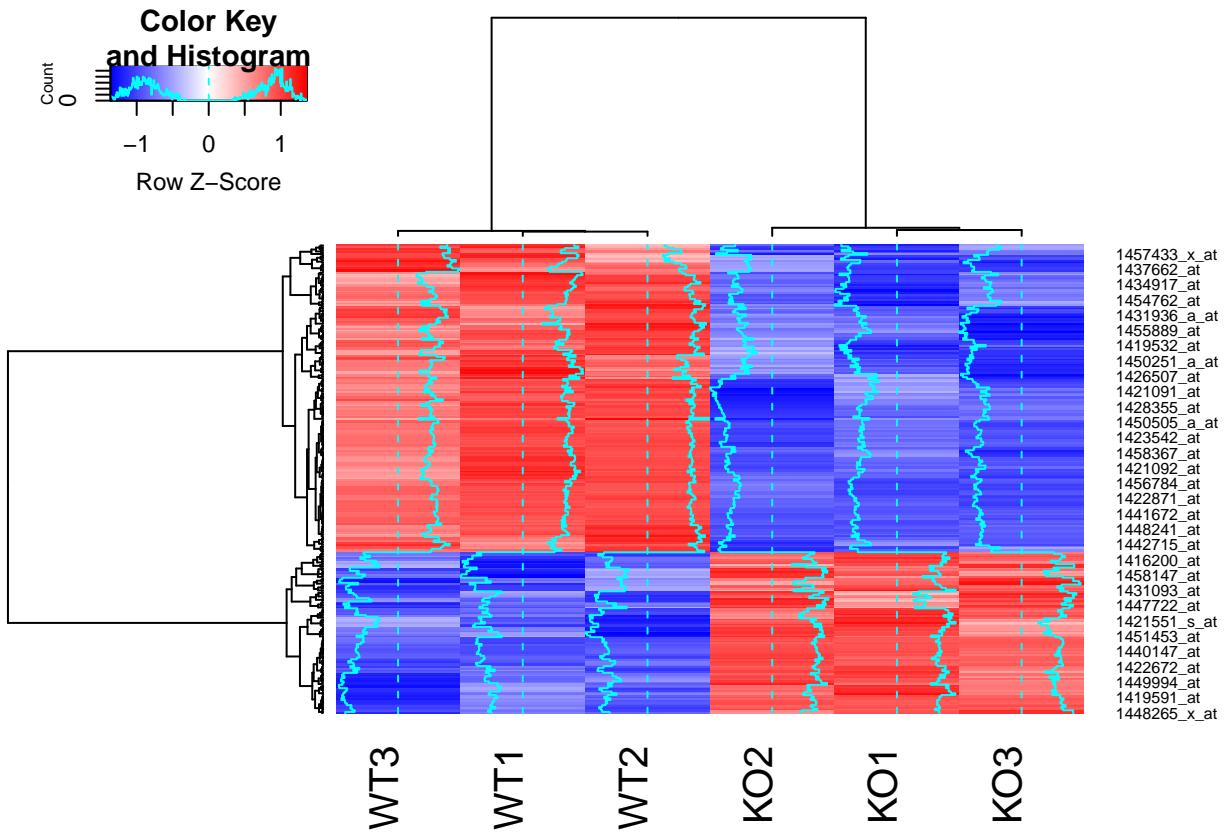
0.46 Enhanced Heatmap 3/4

```
heatmap.2(filtered, Rowv=row_dendrogram, Colv=col_dendrogram,
           col = bluered(256))
```



0.47 Enhanced Heatmap 4/4

```
heatmap.2(filtered, Rowv=row_dendrogram, Colv=col_dendrogram,
           col = bluered(256), scale = "row")
```



0.48 Annotation 1/3

To obtain the functional annotation of the differentially expressed genes, we are going first to extract their probe ids:

```
filterd_ids = row.names(filtered) # ids of the filtered DE genes
length(filterd_ids)

## [1] 276

head(filterd_ids)

## [1] "1416200_at"   "1416236_a_at" "1417808_at"   "1417932_at"
## [5] "1418050_at"   "1418100_at"
```

Then we can obtain the annotation online via The Database for Annotation, Visualization and Integrated Discovery DAVID.

```
write.table(filterd_ids, file = "filterd_ids.txt", quote = FALSE, row.names = FALSE, col.names = FALSE)
```

Now look for a file called “filterd_ids.txt”, open it, copy and paste the ids into DAVID

0.49 Annotation 2/3

Alternatively, we can generate a comprehensive functional annotation via BioConductor packages `annaffy` and `mouse4302.db`.

To install BioConductor packages (if they are not already installed):

```
install.packages("BiocManager")
BiocManager::install(c("annaffy", "mouse4302.db"))
```

Load the packages and extract annotation of the filtered ids:

```
library(annaffy)
annotation_table = aafTableAnn(filterd_ids, "mouse4302.db")
saveHTML(annotation_table, file="filtered.html")
browseURL("filtered.html")

## Loading required package: Biobase
## Loading required package: BiocGenerics
## Loading required package: parallel
##
## Attaching package: 'BiocGenerics'
## The following objects are masked from 'package:parallel':
## 
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB
## The following objects are masked from 'package:dplyr':
## 
##     combine, intersect, setdiff, union
## The following objects are masked from 'package:stats':
## 
##     IQR, mad, sd, var, xtabs
## The following objects are masked from 'package:base':
## 
##     anyDuplicated, append, as.data.frame, basename, cbind,
##     colnames, dirname, do.call, duplicated, eval, evalq, Filter,
##     Find, get, grep, grepl, intersect, is.unsorted, lapply, Map,
##     mapply, match, mget, order, paste, pmax, pmax.int, pmin,
##     pmin.int, Position, rank, rbind, Reduce, rownames, sapply,
##     setdiff, sort, table, tapply, union, unique, unsplit, which,
##     which.max, which.min
## Welcome to Bioconductor
## 
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
## Loading required package: GO.db
## Loading required package: AnnotationDbi
## Loading required package: stats4
## Loading required package: IRanges
## Loading required package: S4Vectors
##
## Attaching package: 'S4Vectors'
```

```

## The following object is masked from 'package:gplots':
##
##      space

## The following objects are masked from 'package:dplyr':
##
##      first, rename

## The following object is masked from 'package:base':
##
##      expand.grid

##
## Attaching package: 'IRanges'

## The following objects are masked from 'package:dplyr':
##
##      collapse, desc, slice

##
## Attaching package: 'AnnotationDbi'

## The following object is masked from 'package:dplyr':
##
##      select

##
## Loading required package: KEGG.db

##
## KEGG.db contains mappings based on older data because the original
## resource was removed from the public domain before the most
## recent update was produced. This package should now be
## considered deprecated and future versions of Bioconductor may
## not have it available. Users who want more current data are
## encouraged to look at the KEGGREST or reactome.db packages

## Loading required package: mouse4302.db

## Loading required package: org.Mm.eg.db

##
##

## Warning in chkPkgs(chip): The mouse4302.db package does not appear to
## contain annotation data.

## Warning in result_fetch(res@ptr, n = n): SQL statements must be issued with
## dbExecute() or dbSendStatement() instead of dbGetQuery() or dbSendQuery().

## Warning in result_fetch(res@ptr, n = n): SQL statements must be issued with
## dbExecute() or dbSendStatement() instead of dbGetQuery() or dbSendQuery().

## An object of class "aafTable"
## Slot "probeids":
## [1] "1416200_at"
##
## Slot "table":
## $Probe
## An object of class "aafList"

```

```

## [[1]]
## An object of class "aafProbe"
## [1] "1416200_at"
##
##
## $Symbol
## An object of class "aafList"
## [[1]]
## [1] "IL33"
## attr(,"class")
## [1] "aafSymbol"
##
##
## $Description
## An object of class "aafList"
## [[1]]
## [1] "interleukin 33"
## attr(,"class")
## [1] "aafDescription"
##
##
## $Chromosome
## An object of class "aafList"
## [[1]]
## [1] "19"
## attr(,"class")
## [1] "aafChromosome"
##
##
## `$Chromosome Location`
## An object of class "aafList"
## [[1]]
## An object of class "aafChromLoc"
## [1] 29925112 29945789
##
##
## $GenBank
## An object of class "aafList"
## [[1]]
## [1] "NM_133775"
## attr(,"class")
## [1] "aafGenBank"
##
##
## $Gene
## An object of class "aafList"
## [[1]]
## An object of class "aafLocusLink"
## [1] 77125
##
##
## $UniGene
## An object of class "aafList"
## [[1]]

```

```

## [1] "Mm.182359"
## attr(,"class")
## [1] "aafUniGene"
##
##
## $PubMed
## An object of class "aafList"
## [[1]]
## An object of class "aafPubMed"
## [1] 8889548 10349636 11042159 11076861 11217851 12466851 12477932
## [8] 12819012 15475267 15489334 16141072 16141073 16286016 16602821
## [15] 17185418 17492053 17623648 17675517 17881510 18003919 18023358
## [22] 18250453 18268038 18450470 18552204 18566365 18603409 18667700
## [29] 18799693 18802081 19248109 19451398 19465481 19508382 19553541
## [36] 19559631 19661270 19666510 19684081 19750479 19841166 19892870
## [43] 19933859 19950183 20035719 20042577 20385815 20412815 20427273
## [50] 20501612 20534524 20634488 20689814 20693421 20937871 20939024
## [57] 21190867 21220696 21239713 21239718 21267068 21268000 21281751
## [64] 21308681 21349253 21357533 21422152 21454686 21469105 21494550
## [71] 21515798 21557213 21642589 21646790 21677750 21703183 21703403
## [78] 21734074 21797940 21835205 21873635 21887788 21945667 21949025
## [85] 21949094 21972019 22013230 22013485 22119406 22142849 22180658
## [92] 22198948 22258632 22267218 22270365 22294690 22307629 22323740
## [99] 22329990 22331917 22370606 22371395 22460070 22542450 22585447
## [106] 22634619 22660580 22661085 22665806 22686327 22689946 22702477
## [113] 22782692 22802353 22922818 23006545 23071771 23093619 23132931
## [120] 23148283 23162017 23169007 23248269 23300625 23323935 23324173
## [127] 23347081 23363980 23397250 23403558 23418608 23496815 23499895
## [134] 23523996 23547117 23582173 23585480 23630360 23662055 23683462
## [141] 23733876 23810766 23837438 23892028 23894196 23911389 23918359
## [148] 23945235 23954132 23960191 24028396 24043894 24045639 24058536
## [155] 24076135 24076431 24105680 24194600 24205109 24220317 24257755
## [162] 24356538 24446518 24459820 24551140 24556514 24586149 24613091
## [169] 24619410 24675360 24730559 24786898 24860117 24860190 24892809
## [176] 24982172 24985397 25015831 25022964 25043027 25143444 25153903
## [183] 25172162 25278425 25313073 25429071 25458701 25472995 25500143
## [190] 25504587 25505285 25533952 25543045 25573803 25599561 25617223
## [197] 25617473 25660244 25661185 25676669 25683166 25692703 25693767
## [204] 25714839 25714983 25739051 25746970 25786179 25795135 25807992
## [211] 25808546 25814531 25829541 25847973 25857925 25870243 25918379
## [218] 25926677 25930197 25941360 25944738 25997709 26011644 26018806
## [225] 26044350 26047701 26079807 26092469 26100084 26129648 26200013
## [232] 26230091 26243875 26244295 26249267 26251474 26272855 26277897
## [239] 26310268 26322482 26342029 26352378 26363151 26365875 26386119
## [246] 26417439 26425820 26428949 26473724 26489077 26490658 26514775
## [253] 26518437 26566691 26597162 26598236 26602156 26603638 26673140
## [260] 26755704 26771472 26802241 26811463 26813347 26830114 26864308
## [267] 26872602 26872699 26908008 26943125 26987428 26991049 27001956
## [274] 27053610 27056266 27067050 27091974 27102638 27120577 27126934
## [281] 27155324 27184849 27222019 27222477 27233150 27236500 27300306
## [288] 27310495 27317338 27318792 27340126 27358001 27372570 27421703
## [295] 27421705 27453471 27492144 27545879 27548066 27559047 27608599
## [302] 27626380 27653694 27683753 27697499 27701734 27749840 27775548
## [309] 27811142 27821781 27839630 27918564 27994070 28011865 28095415

```

```

## [316] 28096407 28137851 28196763 28196875 28198366 28228256 28258042
## [323] 28259547 28341741 28359899 28373582 28389473 28401938 28423665
## [330] 28439013 28448566 28448579 28450225 28494352 28576981 28607531
## [337] 28637902 28637954 28656964 28675392 28697404 28701507 28710436
## [344] 28730605 28771101 28802996 28889151 28921511 29032512 29045819
## [351] 29045902 29045903 29146574 29147584 29166590 29208683 29222107
## [358] 29242561 29247993 29288203 29309756 29368135 29379874 29392836
## [365] 29420261 29452935 29463593 29467297 29534857 29610934 29621782
## [372] 29671873 29674622 29728120 29728508 29729112 29906552 29950152
## [379] 29954866 30001716 30098206 30224451 30539029 30575775 30602786
## [386] 30642984 30763587
##
##
## $`Gene Ontology`
## An object of class "aafList"
## [[1]]
## An object of class "aafGO"
## [[1]][[1]]
## An object of class "aafGOItem"
## @id "GO:0002281"
## @name "macrophage activation involved in immune response"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[2]]
## An object of class "aafGOItem"
## @id "GO:0002282"
## @name "microglial cell activation involved in immune response"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[3]]
## An object of class "aafGOItem"
## @id "GO:0002686"
## @name "negative regulation of leukocyte migration"
## @type "Biological Process"
## @evid "IGI"
##
## [[1]][[4]]
## An object of class "aafGOItem"
## @id "GO:0002826"
## @name "negative regulation of T-helper 1 type immune response"
## @type "Biological Process"
## @evid "IGI"
##
## [[1]][[5]]
## An object of class "aafGOItem"
## @id "GO:0002830"
## @name "positive regulation of type 2 immune response"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[6]]
## An object of class "aafGOItem"

```

```

## @id "GO:0002830"
## @name "positive regulation of type 2 immune response"
## @type "Biological Process"
## @evid "IGI"
##
## [[1]][[7]]
## An object of class "aafGOItem"
## @id "GO:0005125"
## @name "cytokine activity"
## @type "Molecular Function"
## @evid "IBA"
##
## [[1]][[8]]
## An object of class "aafGOItem"
## @id "GO:0005125"
## @name "cytokine activity"
## @type "Molecular Function"
## @evid "IDA"
##
## [[1]][[9]]
## An object of class "aafGOItem"
## @id "GO:0005125"
## @name "cytokine activity"
## @type "Molecular Function"
## @evid "IPI"
##
## [[1]][[10]]
## An object of class "aafGOItem"
## @id "GO:0005125"
## @name "cytokine activity"
## @type "Molecular Function"
## @evid "ISO"
##
## [[1]][[11]]
## An object of class "aafGOItem"
## @id "GO:0005576"
## @name "extracellular region"
## @type "Cellular Component"
## @evid "IEA"
##
## [[1]][[12]]
## An object of class "aafGOItem"
## @id "GO:0005615"
## @name "extracellular space"
## @type "Cellular Component"
## @evid "IEA"
##
## [[1]][[13]]
## An object of class "aafGOItem"
## @id "GO:0005634"
## @name "nucleus"
## @type "Cellular Component"
## @evid "ISO"
##

```

```

## [[1]][[14]]
## An object of class "aafGOItem"
## @id    "GO:0005694"
## @name "chromosome"
## @type "Cellular Component"
## @evid "IEA"
##
## [[1]][[15]]
## An object of class "aafGOItem"
## @id    "GO:0010628"
## @name "positive regulation of gene expression"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[16]]
## An object of class "aafGOItem"
## @id    "GO:0031410"
## @name "cytoplasmic vesicle"
## @type "Cellular Component"
## @evid "IEA"
##
## [[1]][[17]]
## An object of class "aafGOItem"
## @id    "GO:0032436"
## @name "positive regulation of proteasomal ubiquitin-dependent protein catabolic process"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[18]]
## An object of class "aafGOItem"
## @id    "GO:0032436"
## @name "positive regulation of proteasomal ubiquitin-dependent protein catabolic process"
## @type "Biological Process"
## @evid "IGI"
##
## [[1]][[19]]
## An object of class "aafGOItem"
## @id    "GO:0032689"
## @name "negative regulation of interferon-gamma production"
## @type "Biological Process"
## @evid "IGI"
##
## [[1]][[20]]
## An object of class "aafGOItem"
## @id    "GO:0032736"
## @name "positive regulation of interleukin-13 production"
## @type "Biological Process"
## @evid "IGI"
##
## [[1]][[21]]
## An object of class "aafGOItem"
## @id    "GO:0032753"
## @name "positive regulation of interleukin-4 production"
## @type "Biological Process"

```

```

## @evid "IGI"
##
## [[1]][[22]]
## An object of class "aafGOItem"
## @id   "GO:0032754"
## @name "positive regulation of interleukin-5 production"
## @type "Biological Process"
## @evid "IGI"
##
## [[1]][[23]]
## An object of class "aafGOItem"
## @id   "GO:0032755"
## @name "positive regulation of interleukin-6 production"
## @type "Biological Process"
## @evid "IGI"
##
## [[1]][[24]]
## An object of class "aafGOItem"
## @id   "GO:0043032"
## @name "positive regulation of macrophage activation"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[25]]
## An object of class "aafGOItem"
## @id   "GO:0043032"
## @name "positive regulation of macrophage activation"
## @type "Biological Process"
## @evid "ISO"
##
## [[1]][[26]]
## An object of class "aafGOItem"
## @id   "GO:0045944"
## @name "positive regulation of transcription by RNA polymerase II"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[27]]
## An object of class "aafGOItem"
## @id   "GO:0050729"
## @name "positive regulation of inflammatory response"
## @type "Biological Process"
## @evid "IBA"
##
## [[1]][[28]]
## An object of class "aafGOItem"
## @id   "GO:0050729"
## @name "positive regulation of inflammatory response"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[29]]
## An object of class "aafGOItem"
## @id   "GO:0051024"

```

```

## @name "positive regulation of immunoglobulin secretion"
## @type "Biological Process"
## @evid "IGI"
##
## [[1]][[30]]
## An object of class "aafGOItem"
## @id "GO:0051025"
## @name "negative regulation of immunoglobulin secretion"
## @type "Biological Process"
## @evid "IGI"
##
## [[1]][[31]]
## An object of class "aafGOItem"
## @id "GO:0051607"
## @name "defense response to virus"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[32]]
## An object of class "aafGOItem"
## @id "GO:0061518"
## @name "microglial cell proliferation"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[33]]
## An object of class "aafGOItem"
## @id "GO:0090197"
## @name "positive regulation of chemokine secretion"
## @type "Biological Process"
## @evid "IDA"
##
## [[1]][[34]]
## An object of class "aafGOItem"
## @id "GO:0090197"
## @name "positive regulation of chemokine secretion"
## @type "Biological Process"
## @evid "ISO"
##
## [[1]][[35]]
## An object of class "aafGOItem"
## @id "GO:0097191"
## @name "extrinsic apoptotic signaling pathway"
## @type "Biological Process"
## @evid "IGI"
##
##
##
## $Pathway
## An object of class "aafList"
## [[1]]
## An object of class "aafPathway"
## [[1]][[1]]
## An object of class "aafGOItem"

```

Probe	Symbol	Description	Chromosome	Chromosome Location	GenBank	Gene	UniGene	PubMed	Gene Ontology	Pathway
J416200_at	Il33	interleukin 33	19	29945789, 29925113	NM_133775	77125	Mm.182359	267	macrophage activation involved in immune response microglial cell activation involved in immune response negative regulation of leukocyte migration negative regulation of T-helper 1 type immune response positive regulation of type 2 immune response positive regulation of type 2 immune response cytokine activity cytokine activity cytokine activity extracellular region extracellular space nucleus nucleoplasm chromosome cytosol transcription, DNA-templated positive regulation of gene expression cytoplasmic vesicle positive regulation of proteasomal ubiquitin-dependent protein catabolic process positive regulation of proteasomal ubiquitin-dependent protein catabolic process negative regulation of interferon-gamma production positive regulation of interleukin-13 production positive regulation of interleukin-4 production positive regulation of interleukin-5 production positive regulation of interleukin-6	Cytosolic DNA-sensing pathway

Figure 3: Annotation

```
## @id      "04623"
## @name    "Cytosolic DNA-sensing pathway"
## @enzyme  ""
```

0.50 Annnotation 3/3

0.51 Exercise

Generate separate annotations for the up-regulated and down-regulated differentially expressed genes between WT and KO. Discuss each set in a molecular context as detailed as possible.

0.52 Homework

- Identify the top 10 *biologically* significant genes (i.e., by fold-change)
- Identify the top 10 *statistically* significant genes (i.e., by *p*-value)
- Identify the overlapping (if any) between the two sets of genes