Statistical Analysis of RNA-Seq Data

Minghui Wang, Qi Sun

Bioinformatics Facility
Cornell University

Gene count table

Genes	Condition A	Condition B
Gene A	10	30
Gene B	30	90
Gene C	5	15
Gene D	1	3
Gene N	80	240
	126	378

Genes	Condition A	Condition B
Gene A	10	5
Gene B	30	60
Gene C	5	1
Gene D	1	1
Gene N	80	59
	126	126

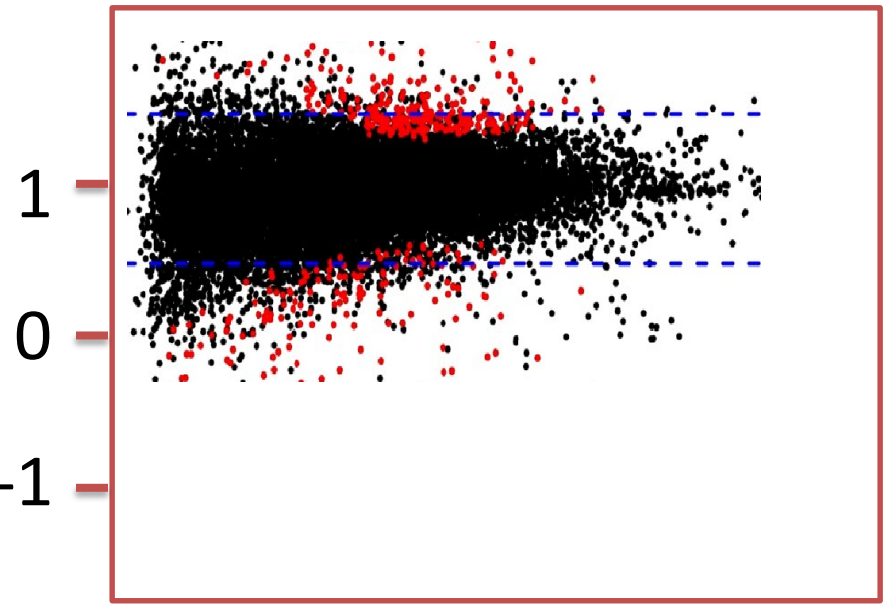
- Library size
- RNA composition bias

Normalization

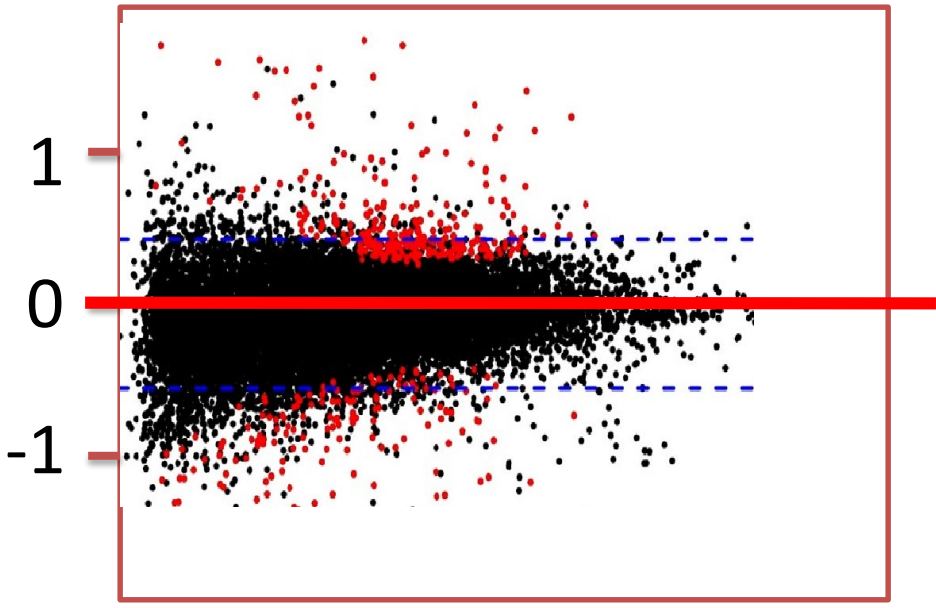
Is necessary in RNAseq as total read counts are different in different samples

MA Plots

Before normalization



After normalization



- Y axis: log ratio of expression level between two conditions;
- With the assumption that most genes are expressed equally, the log ratio should mostly be close to 0

Normalization methods

- **Total-count normalization**
  - By total mapped reads
- **Upper-quantile normalization**
  - By read count of the gene at upper-quantile
- **Normalization by housekeeping genes**
- **Trimmed mean (TMM) normalization**

Normalization methods

- **Total-count normalization (FPKM, RPKM)**
  - By total mapped reads (in transcripts)
- **Upper-quartile normalization**
  - By read count of the gene at upper-quartile
- **Normalization by housekeeping genes**
- **Trimmed mean (TMM) normalization**

Default

cuffdiff

EdgeR

A simple normalization

FPKM (CUFFLINKS)

Fragments Per Kilobase Of Exon Per Million Fragments

Normalization factor:

Default: total reads from genes defined in GFF

-total-hits-norm: all aligned reads

CPM (EdgeR)

Count Per Million Reads

Normalization factor:

- total reads from genes defined in GFF
- Correction with TMM

Reads that are not mapped to gene region (e.g. rRNA, pseudo-genes would not affect normalization

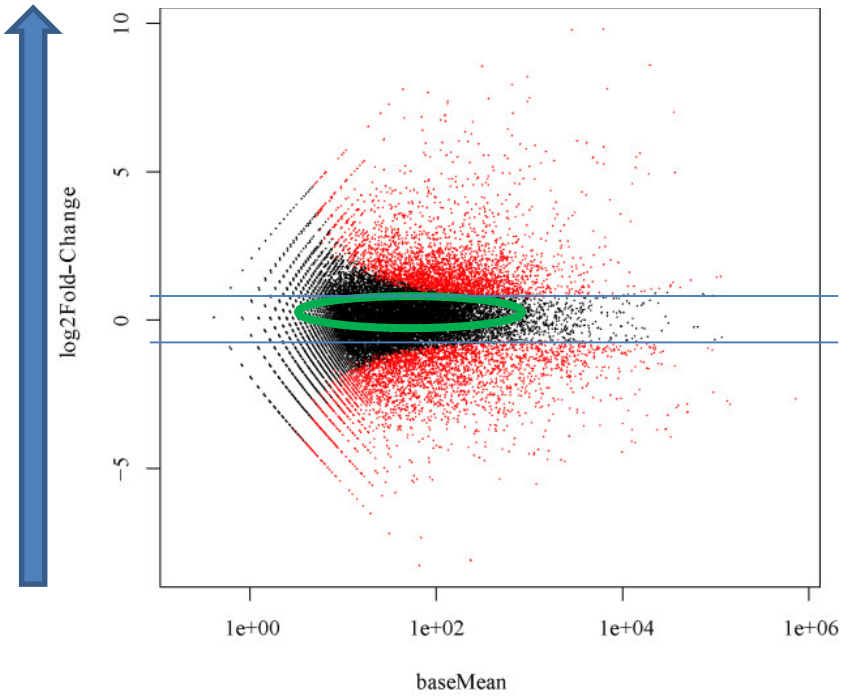
TMM normalization step

Gene	Sample 1	Sample 2	Sample 3	Sample 4
GENE A	10	10	10	30
GENE B	30	20	10	80
GENE C	20	30	20	20
GENE D	50	30	40	30
GENE E	30	20	30	10
	140	110	110	170

Gene	Sample 1	Sample 2	Sample 3	Sample 4
GENE A	0.07	0.09	0.09	0.18
GENE B	0.21	0.18	0.09	0.47
GENE C	0.14	0.27	0.18	0.12
GENE D	0.36	0.27	0.36	0.18
GENE E	0.21	0.18	0.27	0.06
Q (75%)	**0.21**	0.27	0.27	0.18
Mean				**0.23**

TMM normalization step

Gene	Ref	Test	M	A
GENE 1	10	20	…	…
GENE 2	20	30	…	…
GENE 3	50	20	…	…
GENE 4	30	20	…	…
GENE 5	30	30	…	…
GENE 6	10	50	…	…



M = log2(Test/Test_total)-log2(Ref/Ref_total)

A =0.5 *log2(Test/Test_total*Ref/Ref_total)

Effective library size

1 1.02 0.99 1.1

DESeq2 normalization

A

Gene	A	B	C
Gene 1	0	1	20
Gene 2	2	3	5
Gene 3	4	10	100

B

Gene	A	B	C
Gene 1	inf	0	3
Gene 2	0.69	1.1	1.6
Gene 3	1.38	2.3	4.6

inf
1.13
2.76

C

Gene	A	B	C
Gene 1	inf	0	3
Gene 2	-0.44	-0.03	0.47
Gene 3	-1.38	-0.46	1.84

D

Gene	A	B	C
Gene 1	inf	0	3
Gene 2	-0.44	-0.03	0.47
Gene 3	-1.38	-0.46	1.84

E **-0.91** **-0.245** **1.16**

F **0.40** **0.78** **3.19**

3. Differentially expressed genes

Given a gene:

Read counts in control samples:

Repeat 1 **24**

Repeat 2 **25**

Repeat 3 **27**

Read counts in treated samples:

Repeat 1 **23**

Repeat 2 **47**

Repeat 3 **29**

Different statistics model might give you different P or Q values.

Comparison of Methods

Table 2

Comparison of methods.

Evaluation	Cuffdiff	DESeq	edgeR	limmaVoom	PoissonSeq	baySeq
Normalization and clustering	All methods performed equally well					
DE detection accuracy measured by AUC at increasing qRT-PCR cutoff	Decreasing	Consistent	Consistent	Decreasing	Increases up to log expression change ≤ 2.0	Consistent
Null model type I error	High number of FPs	Low number of FPs	Low number of FPs	Low Number of FPs	Low number of FPs	Low number of FPs
Signal-to-noise vs *P* value correlation for genes detected in one condition	Poor	Poor	Poor	Good	Moderate	Good
Support for multi-factored experiments	No	Yes	Yes	Yes	No	No
Support DE detection without replicated samples	Yes	Yes	Yes	No	Yes	No
Detection of differential isoforms	Yes	No	No	No	No	No
Runtime for experiments with three to five replicates on a 12 dual-core 3.33 GHz, 100 G RAM server	Hours	Minutes	Minutes	Minutes	Seconds	Hours

AUC, area under curve; DE, differential expression; FP, false positive.

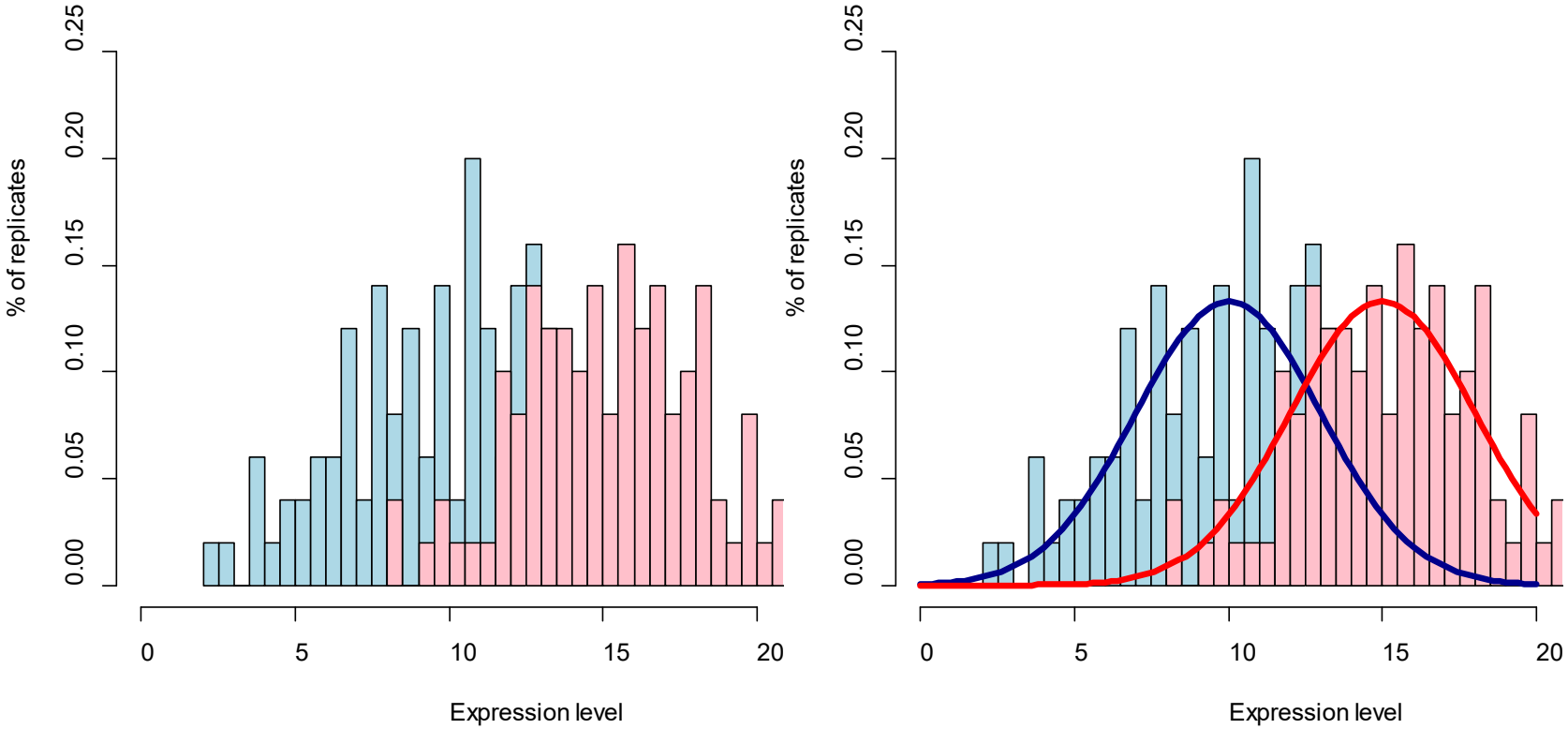
Rapaport *et al. Genome Biology* 2013 **14**:R95 doi:10.1186/gb-2013-14-9-r95

Can I trust P-value?
Can I trust Adjusted P-value?

Rapaport F et al. Genome Biology, 2013 14:R95

3. Differentially expressed genes

If we could do 100 biological replicates,

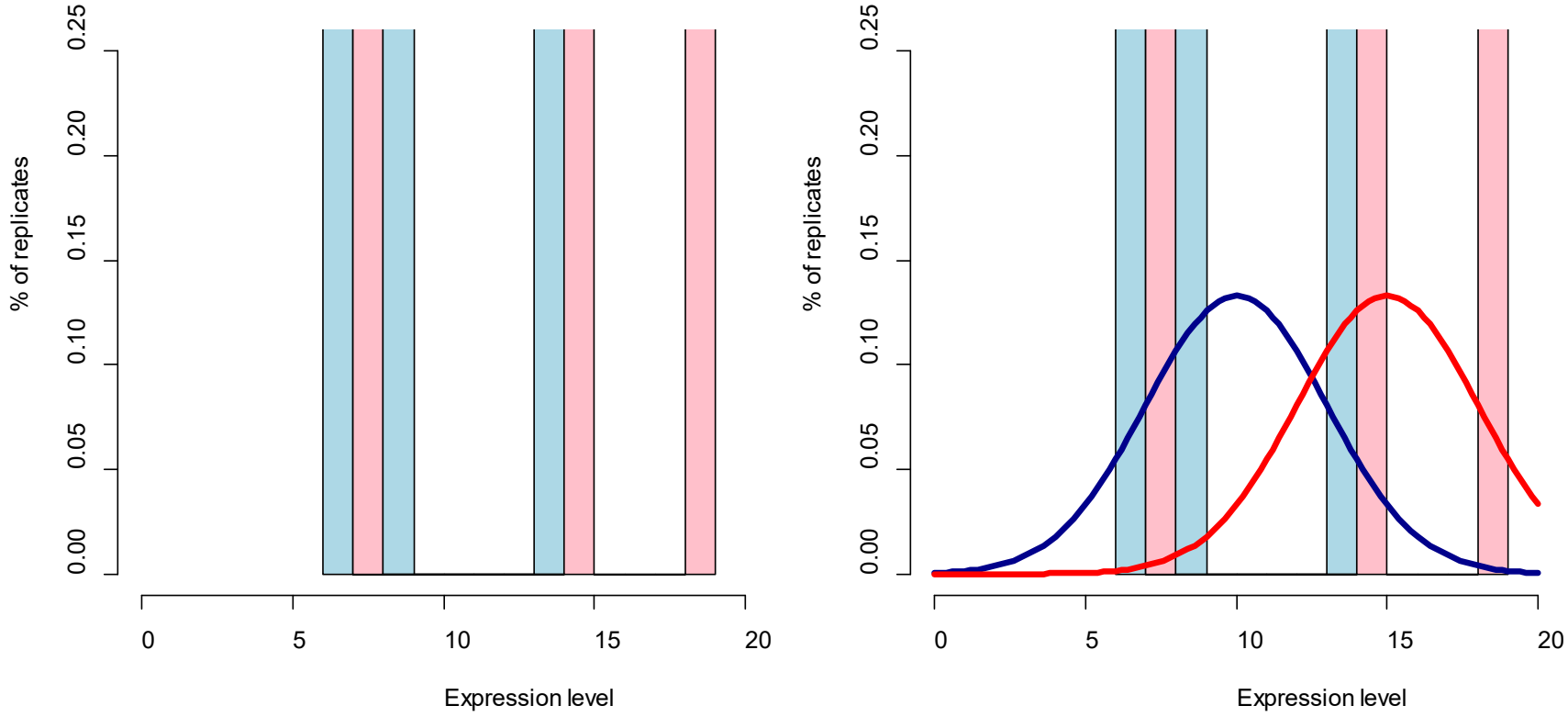


Distribution of Expression Level of A Gene

Condition 1

Condition 2

The reality is, we could only do 3 replicates,



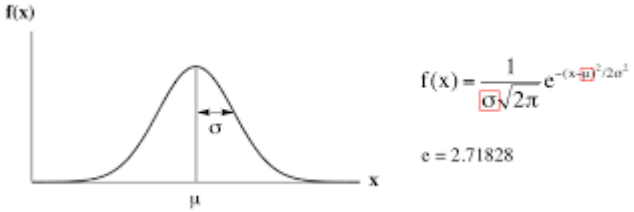
Distribution of Expression Level of A Gene

Condition 1

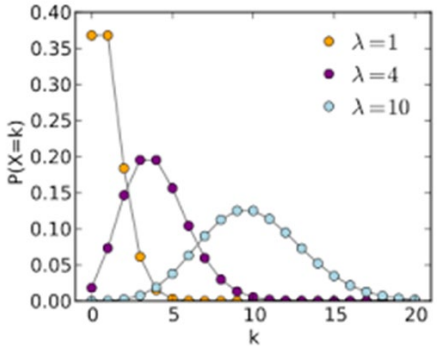
Condition 2

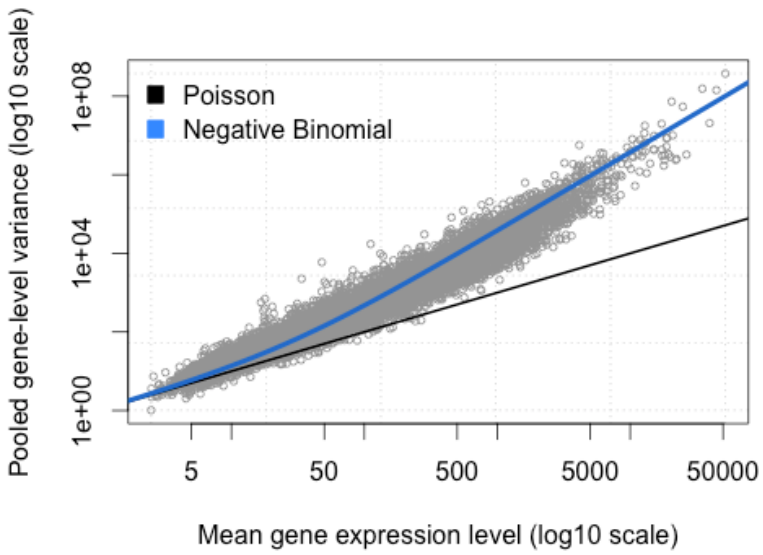
Statistical models

- Gaussian distribution



- Poisson distribution





- Negative binomial distribution

Statistical modeling of gene expression and test for differentially expressed genes

1. Estimate of variance.

Eg. EdgeR uses a combination of

1) a common dispersion effect from all genes;
2) a gene-specific dispersion effect.

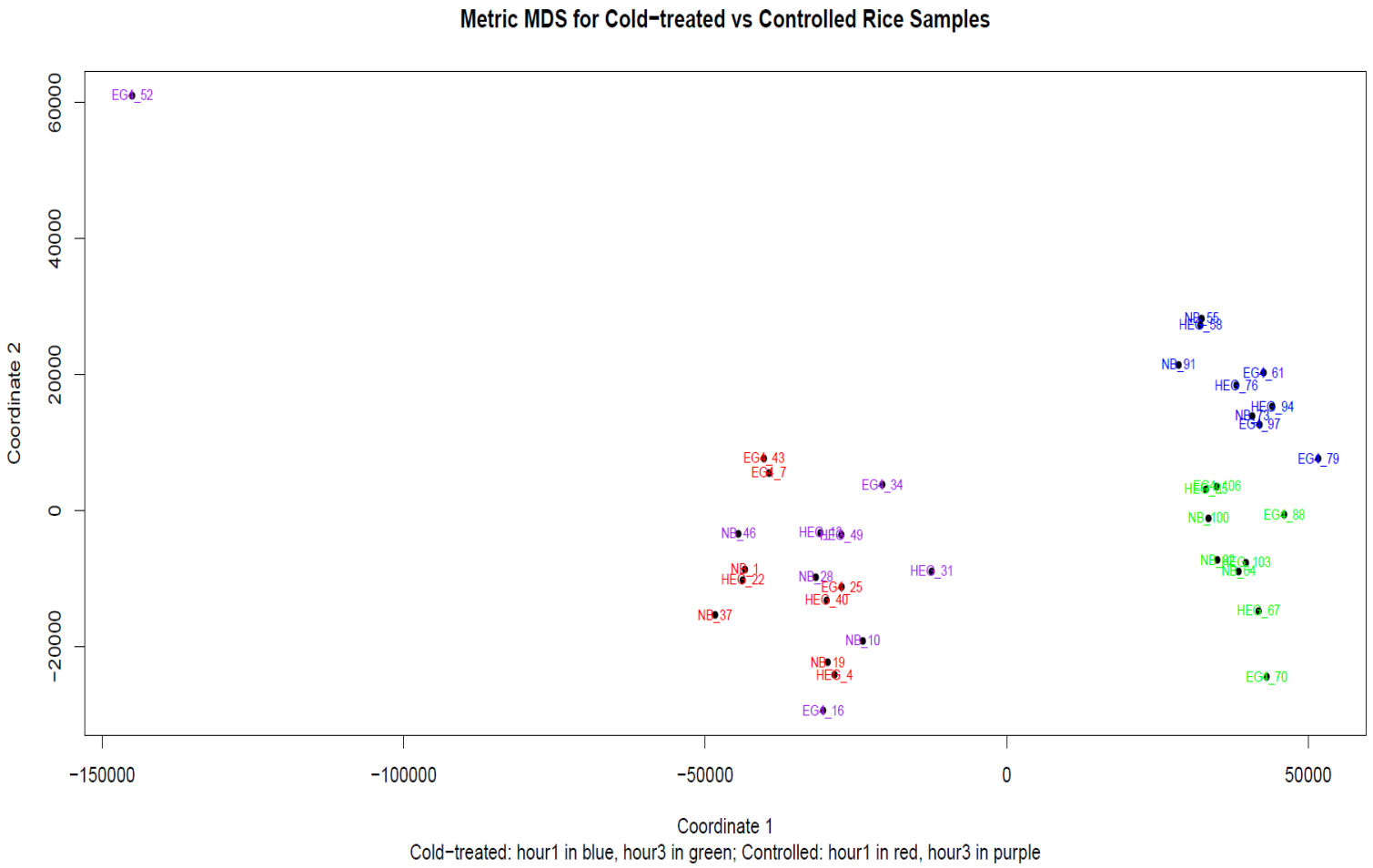
2. Model the expression level with negative bionomial distribution.

DESeq and EdgeR

3. Multiple test correction

Default in EdgeR: Benjamini-Hochberg

Using EdgeR to make MDS plot of the samples



- Check reproducibility from replicates, remove outliers
- Check batch effects;

Output table from RNA-seq pipeline

Values for each gene:

- Read count (raw & normalized)
- Fold change (Log2 fold) between the two conditions
- P-value
- Q(FDR) value after multiple test.

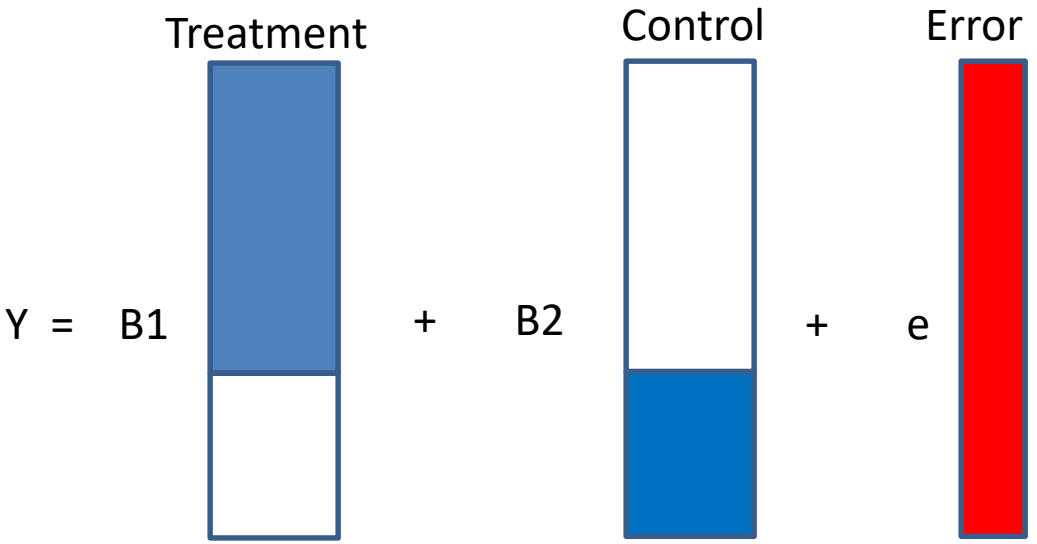
Filter by:

a. **fold change;**
b. **FDR value to filter;**
c. **Expression level.**

E.g. Log2(fold) >1 or <-1
FDR < 0.05

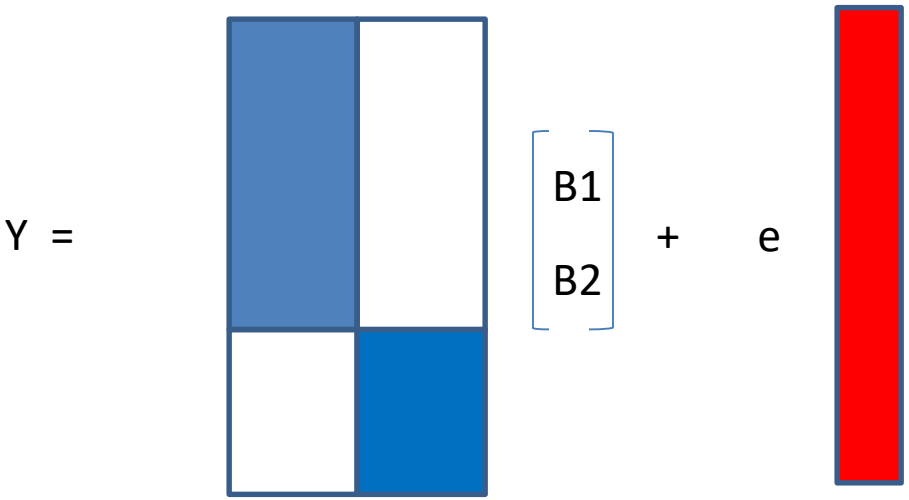
Design Vs Contrast matrix

data	Value
Sample 1	10
Sample 2	15
Sample 3	11
Sample 4	23
Sample 5	11



$Y = B1*X1+B2*X2+e$

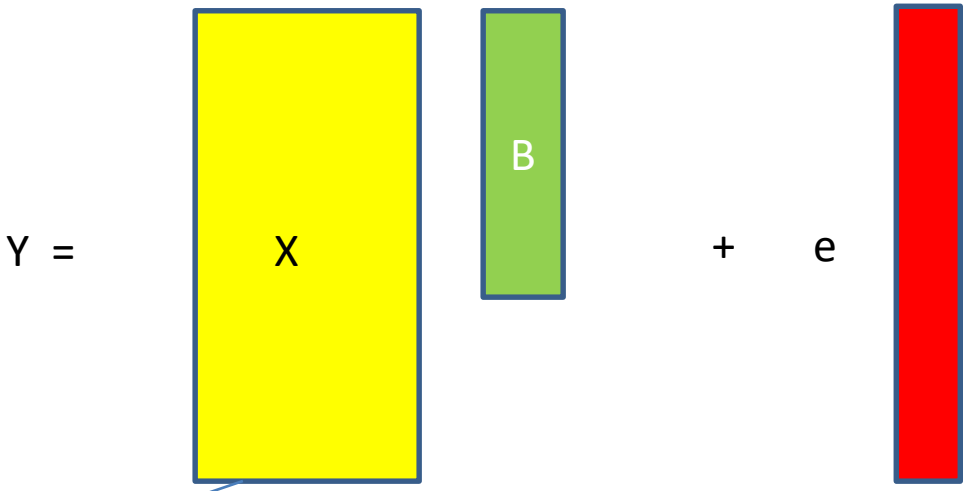
data	Value
Sample 1	10
Sample 2	15
Sample 3	11
Sample 4	23
Sample 5	11



$Y = XB + e$

Design Vs Contrast matrix

data	Value
Sample 1	10
Sample 2	15
Sample 3	11
Sample 4	23
Sample 5	11



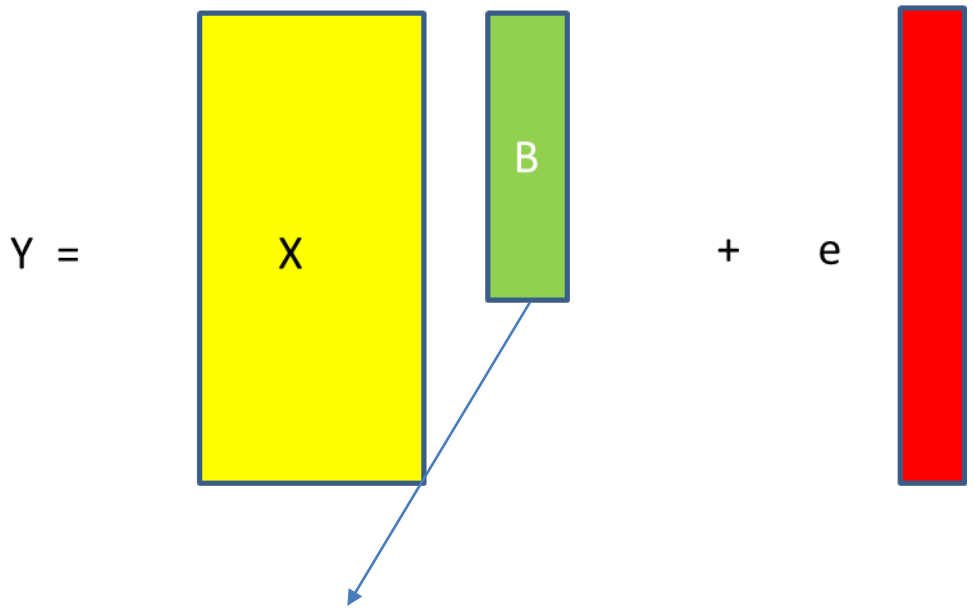
Design matrix All available factors related to performed experiments and potential confounds

```
group <-c("T","T","T","C","C")
design<-model.matrix(~0+group)
design
groupC groupT
     0      1
     0      1
     0      1
     1      0
     1      0
```

```
group <-c("T","T","T","C","C","D","D","H","H","H")
design<-model.matrix(~0+group)
design
 groupC groupD groupH groupT
      0      0      0      1
      0      0      0      1
      0      0      0      1
      1      0      0      0
      1      0      0      0
      0      1      0      0
      0      1      0      0
      0      0      1      0
      0      0      1      0
      0      0      1      0
```

Design Vs Contrast matrix

data	Value
Sample 1	10
Sample 2	15
Sample 3	11
Sample 4	23
Sample 5	11



Contrast matrix Comparing effects of interest and perform statistical evaluation of hypotheses.

C’ B where C’ = [1,-1]

C’B =[1,-1] [B1,B2]’ =1*B1+ (-1)*B2 **? 0**

Attention The Contrast matrix is depend on the design of experiment

Connection between software

Reading file into R

AT1G01010	57	49	36	40
AT1G01020	172	148	197	187
AT1G03987	0	0	0	0
AT1G01030	88	77	74	101
AT1G01040	594	669	504	633
AT1G03993	2	1	0	0
…	…	…	…	…

```
x <- read.delim("gene_count.txt", header=F, row.names=1)

colnames(x)<-c("WTa","WTb","MUa","MUb")
```

Use EdgeR to identify DE genes

	Treat	Time
Sample 1-3	Drug	0 hr
Sample 4-6	Drug	1 hr
Sample 7-9	Drug	2 hr

Normalization and Remove genes that are not expressed

```
library("edgeR")
group <- factor(c(1,1,2,2))
y <- DGEList(counts=x,group=group)
y <- calcNormFactors(y)
keep <-rowSums(cpm(y)>=1) >=2   # remove un-expressed genes
y<-y[keep,]
```

Use EdgeR to identify DE genes

	Treat	Time
Sample 1-3	Drug	0 hr
Sample 4-6	Drug	1 hr
Sample 7-9	Drug	2 hr

Fit the model:

```
group <- factor(c(1,1,1,2,2,2,3,3,3))
design <- model.matrix(~0+group)
fit <- glmFit(myData, design)

lrt12 <- glmLRT(fit, contrast=c(1,-1,0))    #compare 0 vs 1h
lrt13 <- glmLRT(fit, contrast=c(1,0,-1))    #compare 0 vs 2h
lrt23 <- glmLRT(fit, contrast=c(0,1,-1))    #compare 1 vs 2h
```

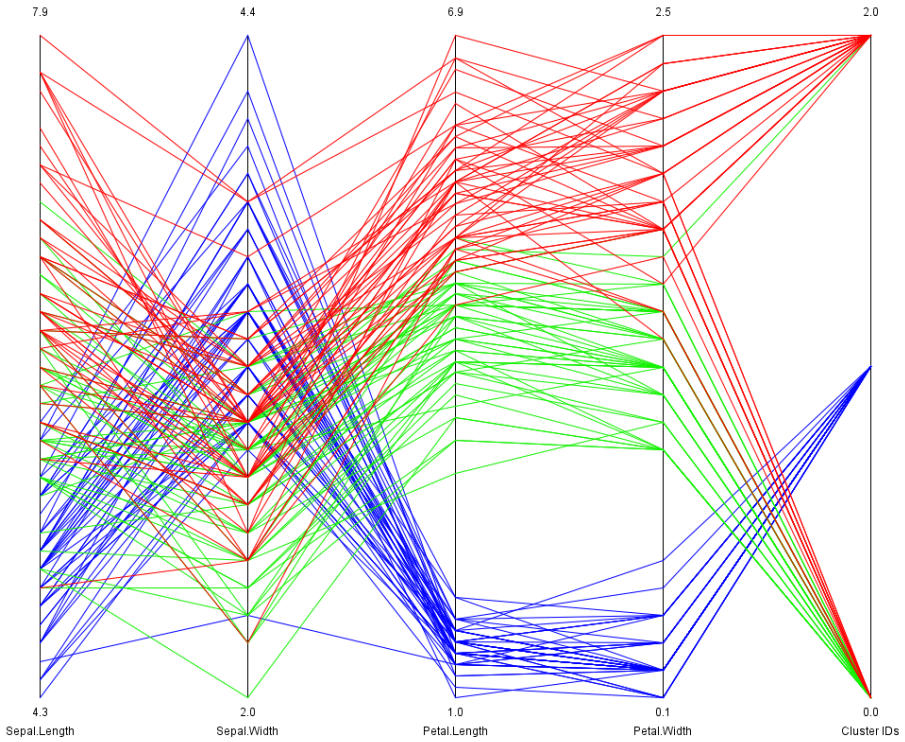
Multiple-factor Analysis in EdgeR

	Treat	Time
Sample 1-3	Placebo	0 hr
Sample 4-6	Placebo	1 hr
Sample 7-9	Placebo	2 hr
Sample 10-12	Drug	0 hr
Sample 13-15	Drug	1 hr
Sample 16-18	Drug	2 hr

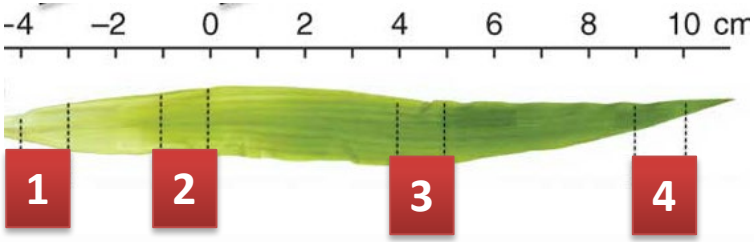
```
group <- factor(c(1,1,1,2,2,2,3,3,3,4,4,4,5,5,5,6,6,6))
design <- model.matrix(~0+group)
fit <- glmFit(mydata, design)

lrt <- glmLRT(fit, contrast=c(-1,0,1,1,0,-1))
### equivalent to (Placebo.2hr – Placbo.0hr) – (Drug.2hr-
Drug.1hr)
```

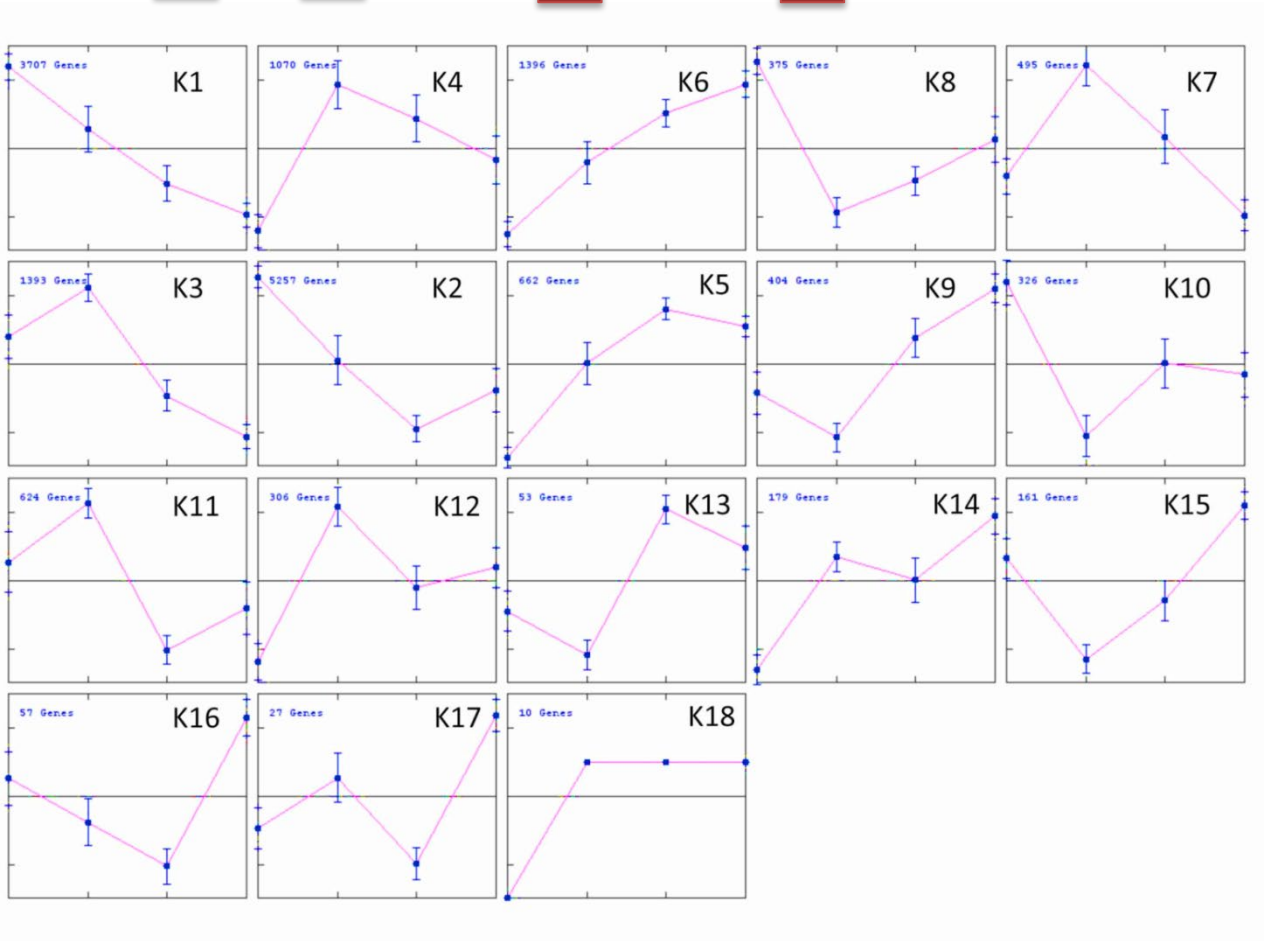
Clustering analysis

1. Hierarchical
2. K-means
3. Co-expression network



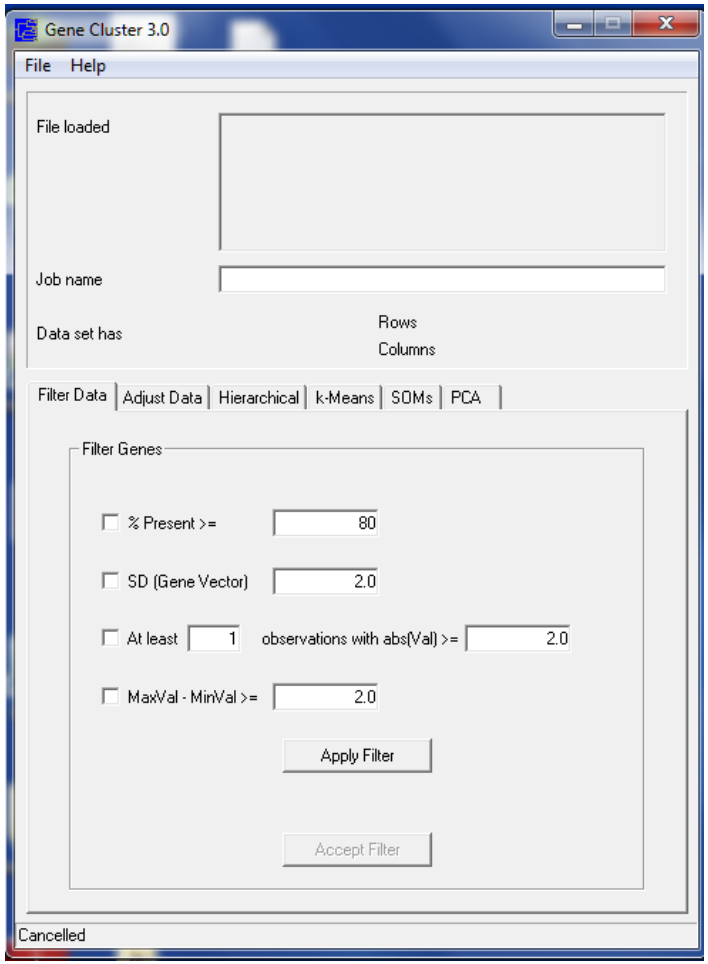
Clustering analysis on multiple conditions of RNA-seq data





Using free software Cluster 3.0 for hierarchical and k-means clustering

http://bonsai.hgc.jp/~mdehoon/software/cluster/software.htm

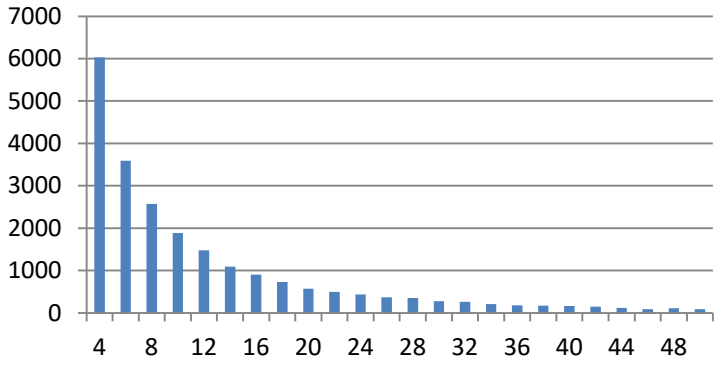


* Add 1 to each FPKM value before loading into Cluster

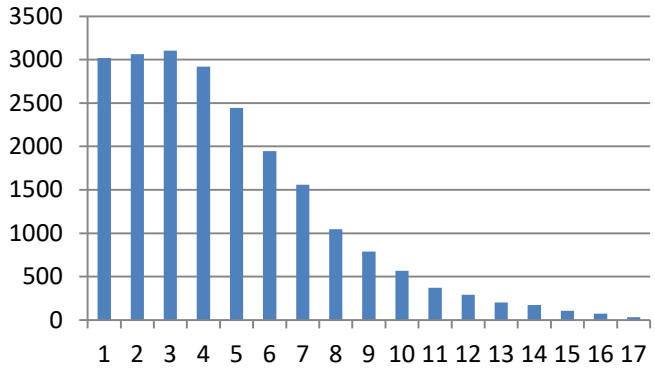
tracking_id	s1_FPKM	s2_FPKM	s3_FPKM	s4_FPKM
AC148152.3_FG001	1	1	1.085823	1.237447
AC148152.3_FG002	1	1	1	1
AC148152.3_FG005	1.054317	6.65432	1.089866	1
AC148152.3_FG006	1.044314	1.223353	1	1
AC148152.3_FG007	1	1	1	1
AC148152.3_FG008	3.13339	20.1778	68.1838	88.5417
AC148167.6_FG001	17.603	43.4081	54.7869	37.5133
AC149475.2_FG002	149.468	10.75707	14.3301	11.8052
AC149475.2_FG003	101.308	34.2556	30.6524	20.2889
AC149475.2_FG004	1.053882	1	1	1

Prepare data for clustering

Step 1. LOG transformation of CPM value to improve the distribution



CPM



Log2(CPM)

To Avoid log(0), using Excel to add 1 to all FPKM values before loading to Cluster.

Prepare data for clustering

Step 2. Filter data

Remove

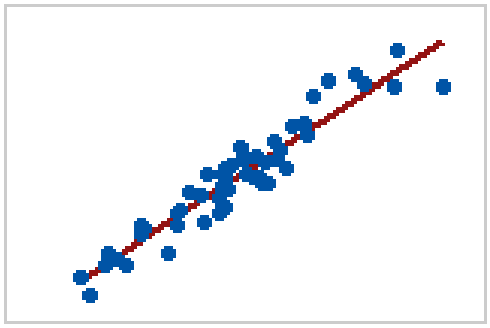
- **Low expressed genes;**
- **Invariant genes.**

Pairwise distance matrix of all genes

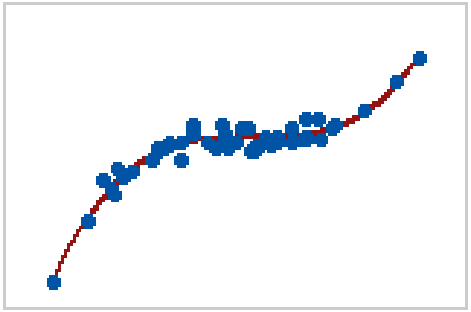
Pearson : Linear correlation (Default)

vs

Spearman: Ranked correlation



Use Pearson



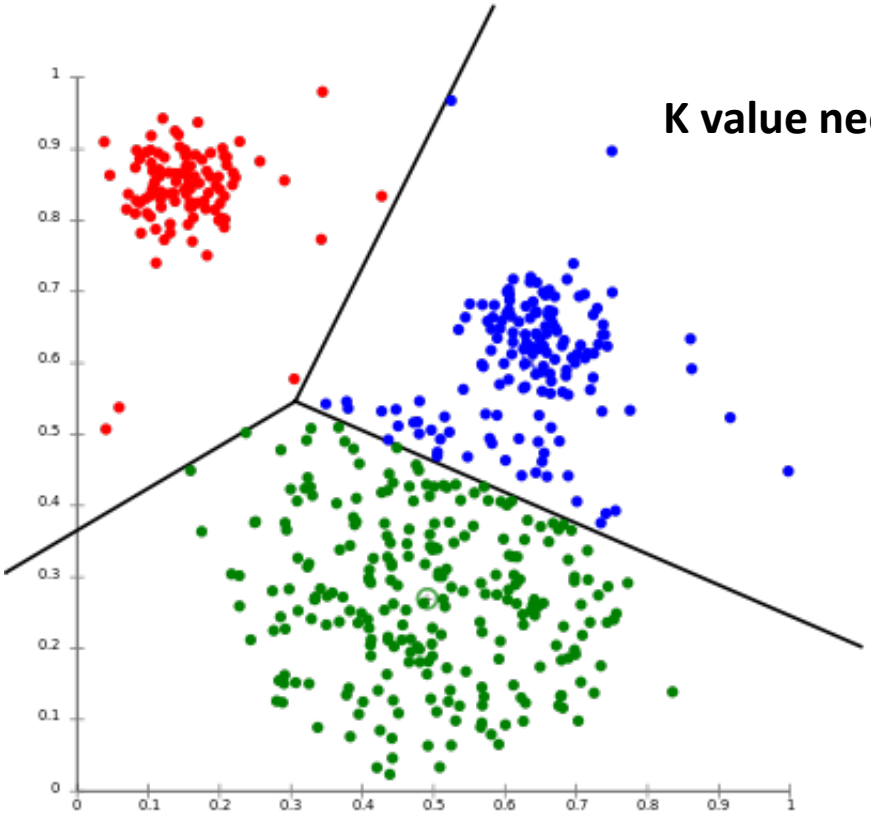
Use Spearman

Hierarchical clustering

Visualize the clustering results with Treeview

The software has functions to select nodes and export genes in selected node.

K-means clustering

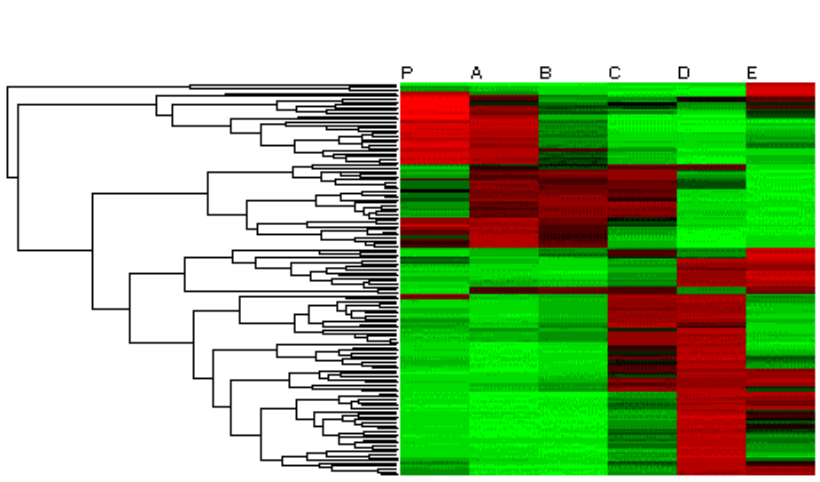


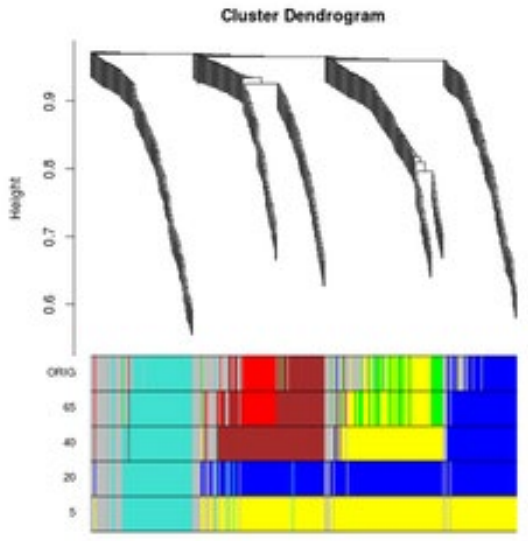
K value need to be pre-specified

Co-expression network modules

WGCNA (weighted correlation network analysis)

- transform the initial distance matrix into Topological Overlap Matrix





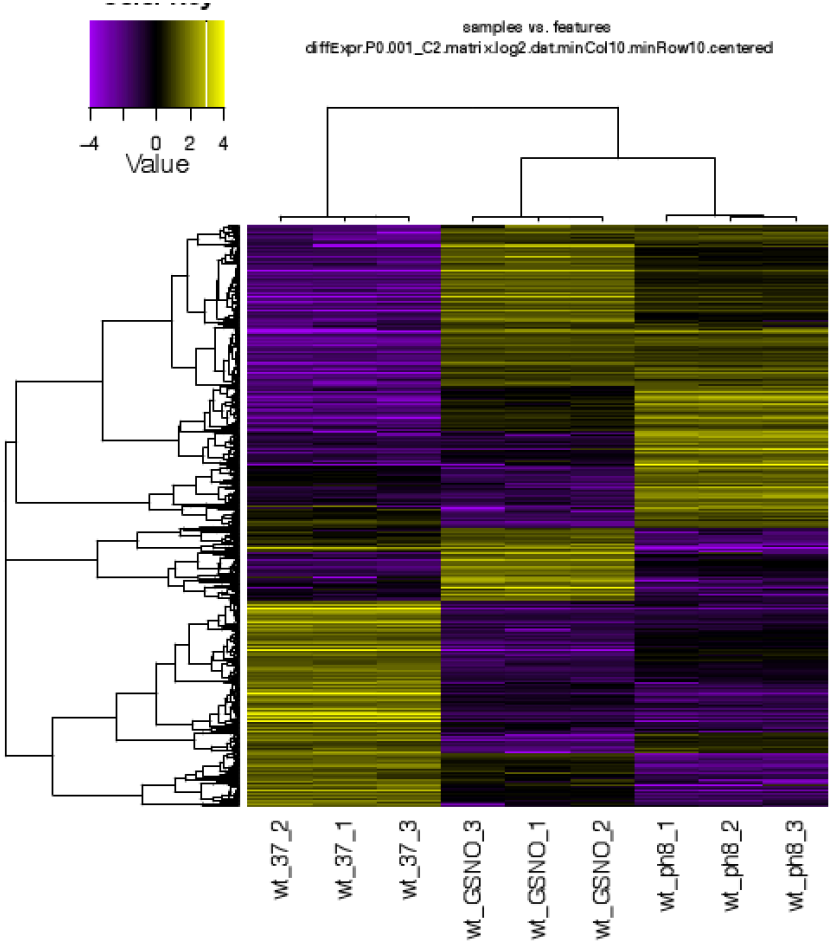
http://rgm3.lab.nig.ac.jp/RGM/R_image_list?package=WGCNA&init=true

Alternative software

- **Bioconductor:**
  - **hclust**
  - **kmeans**

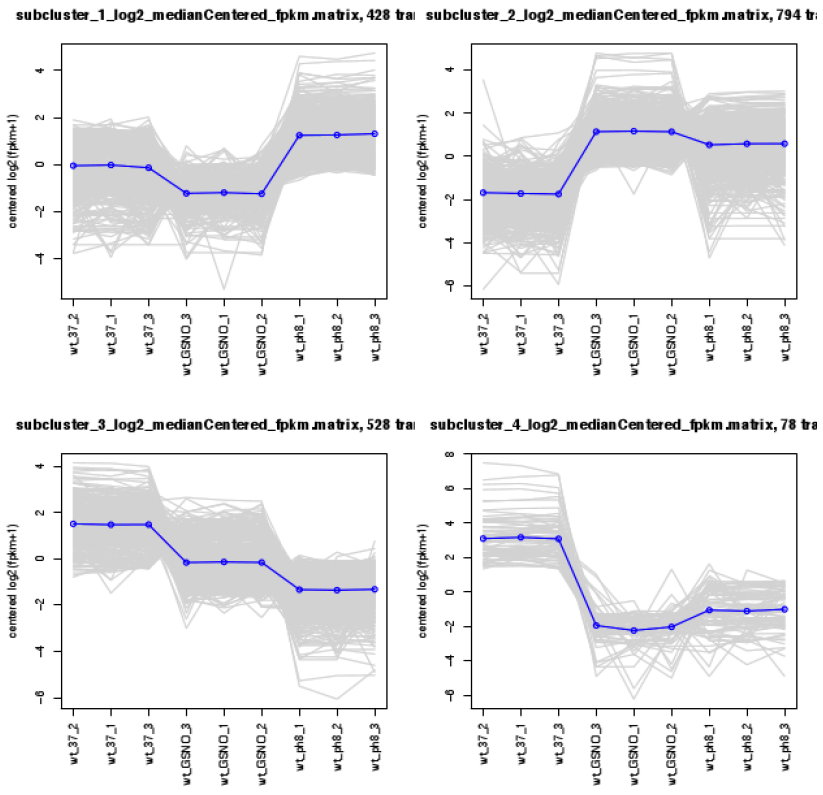
Using scripts in Trinity Package

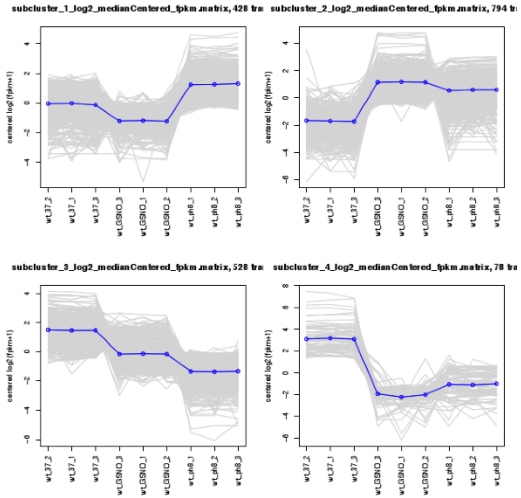
```
$TRINITY_HOME/Analysis/DifferentialExpression/analyze_diff_expr.pl --matrix ./EXPR.matrix -P 1e-3 -C 2
```



K-means clustering

```
$TRINITY_HOME/Analysis/DifferentialExpression/
define_clusters_by_cutting_tree.pl  -R
diffExpr.P0.001_C2.matrix.RData -K 18
```

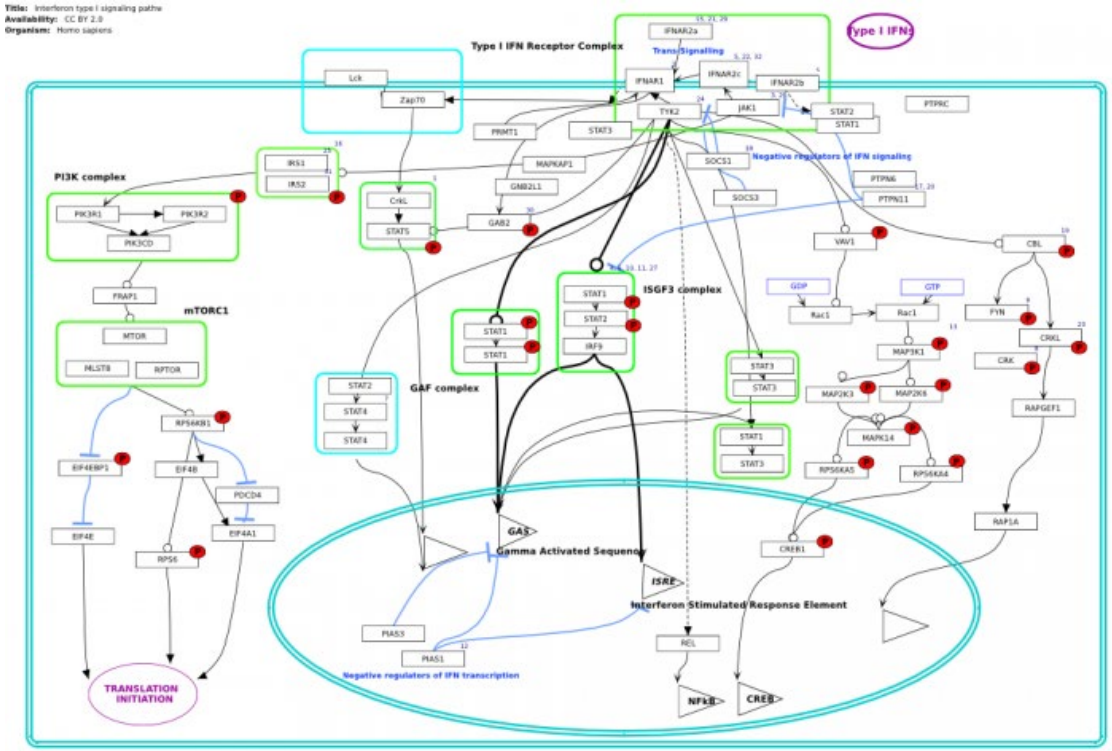




Gene Set Enrichment Analysis

Will be covered in this workshop:

Genome Annotation And Sequence Based Gene Function Prediction (December 12 and 19 2018)



https://www.ebi.ac.uk/training/online/course/functional-genomics-ii-common-technologies-and-data-analysis-methods/gene-set-enrichment