

# **TRGN 527: Applied Data Science and Bioinformatics**

## **UNIT III. Supervised Statistical Tests**

### **Week 7 - Lecture 1 – Case Study Part 3**

**Enrique I. Velazquez Villarreal, M.D., Ph.D., M.P.H., M.S. | Assistant Professor**

Dept. of Translational Genomics

USC | Keck School of Medicine | Norris Comprehensive Cancer Center

Leader of the USC Bioinformatics Core – *USC CaRE2 Health Equity Center*

**David W. Craig, Ph.D. | Professor and Vice Chair**

Dept. of Translational Genomics

USC | Keck School of Medicine | Norris Comprehensive Cancer Center

Co-Director, Institute of Translational Genomics

# Differential gene expression

- Most often, the ultimate goal of gene expression microarray experiments is to identify genes that are significantly over- or under-expressed in groups of samples.
- The most often used approaches doing this are implemented in the limma package (Ritchie et al. 2015).
- After the intensities of microarray samples are normalized, it is a good idea to proceed with checking if there are samples grouped together based on all genes in an analysis.

## Generating a PCAs for sample-grouping purposes

### Installing affycoretools package

```
# if (!requireNamespace("BiocManager", quietly = TRUE))
#   install.packages("BiocManager")
# BiocManager::install("affycoretools", version = "3.8")
```

```
library(affycoretools)
```

# Differential gene expression

- The associations of samples can be studied by applying a principal component analysis.
- The plotPCA() function of affycoretools is a painless, quick way to achieve this.

Generating a PCAs for sample-grouping purposes

Installing affycoretools package

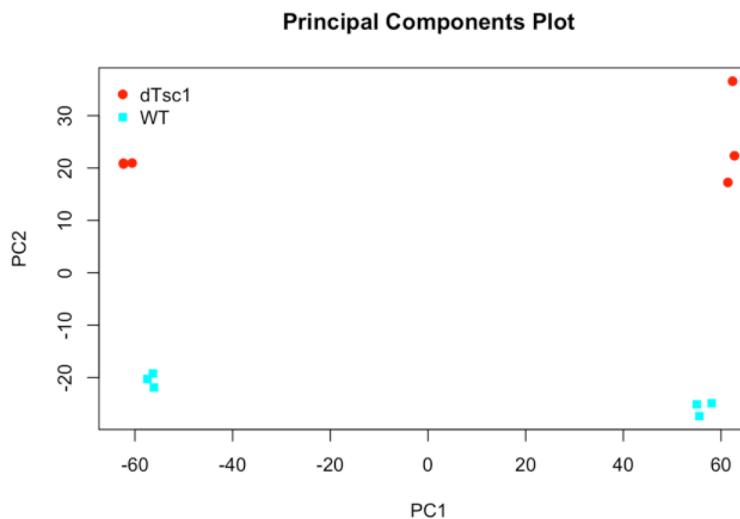
```
# if (!requireNamespace("BiocManager", quietly = TRUE))
#   install.packages("BiocManager")
# BiocManager::install("affycoretools", version = "3.8")

library(affycoretools)

##

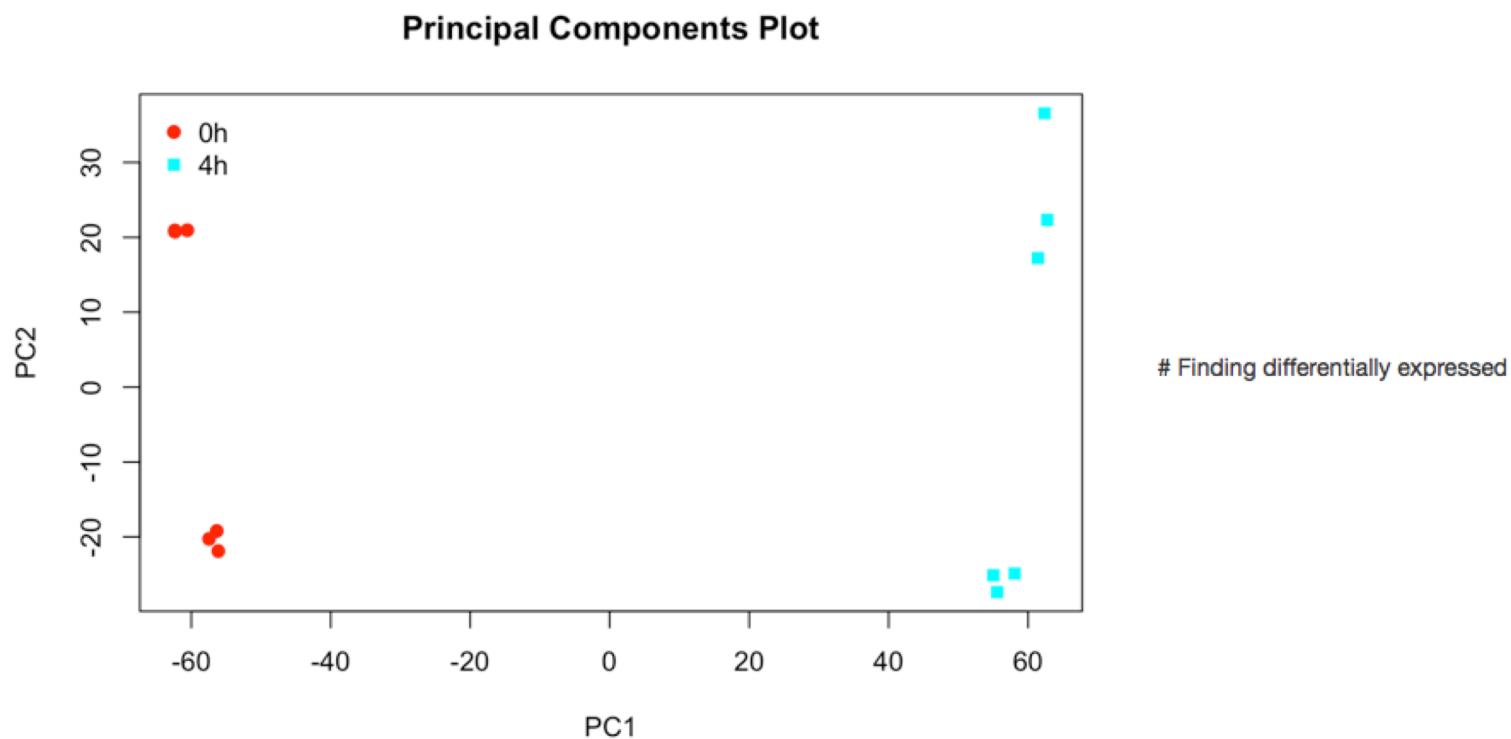
##

plotPCA(TRGN599_matexp, groups = as.numeric(pData(TRGN599_data)[,2]), groupnames = levels(pData(TRGN599_data)[,2])
))
```



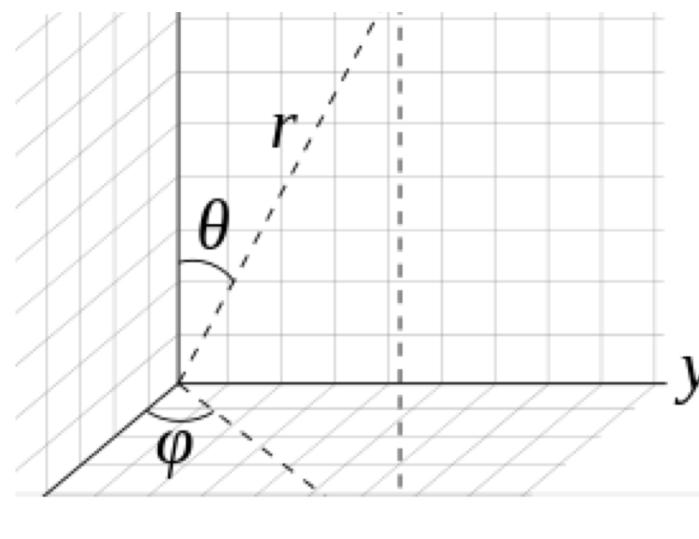
# Differential gene expression

```
plotPCA(TRGN599_matexp, groups = as.numeric(pData(TRGN599_data)[,3]), groupnames = levels(pData(TRGN599_data)[,3]))
```



# Differential gene expression

- Principal component analysis (PCA)
- Statistical procedure that uses an orthogonal transformation (Linear transformation)
- Convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components.
- Mathematically PCA is defined as an orthogonal linear transformation that transforms the data to a new coordinate system
  - First principal component: the greatest variance by some projection of the data that comes to lie on the first coordinate
  - Second principal component: the second greatest variance on the second coordinate
  - ...



# Differential gene expression

## Package ‘limma’

February 20, 2019

**Version** 3.39.12

**Date** 2019-02-15

**Title** Linear Models for Microarray Data

**Description** Data analysis, linear models and differential expression for microarray data.

**Author** Gordon Smyth [cre,aut], Yifang Hu [ctb], Matthew Ritchie [ctb], Jeremy Silver [ctb], James Wettenhall [ctb], Davis McCarthy [ctb], Di Wu [ctb], Wei Shi [ctb], Belinda Phipson [ctb], Aaron Lun [ctb], Natalie Thorne [ctb], Alicia Oshlack [ctb], Carolyn de Graaf [ctb], Yunshun Chen [ctb], Mette Langaas [ctb], Egil Ferkningstad [ctb], Marcus Davy [ctb], Francois Pepin [ctb], Dongseok Choi [ctb]

**Maintainer** Gordon Smyth <[smyth@wehi.edu.au](mailto:smyth@wehi.edu.au)>

**License** GPL (>=2)

**Depends** R (>= 2.3.0)

**Imports** grDevices, graphics, stats, utils, methods

**Suggests** affy, AnnotationDbi, BiasedUrn, Biobase, ellipse, GO.db, gplots, illuminaio, locfit, MASS, org.Hs.eg.db, splines, statmod (>= 1.2.2), vsn

**URL** <http://bioinf.wehi.edu.au/limma>

# Differential gene expression

## R topics documented:

01.Introduction . . . . .	5
02.Classes . . . . .	6
03.ReadingData . . . . .	7
04.Background . . . . .	8
05.Normalization . . . . .	9
06.LinearModels . . . . .	10
07.SingleChannel . . . . .	12
08.Tests . . . . .	12
09.Diagnostics . . . . .	13
10.GeneSetTests . . . . .	14
11.RNAseq . . . . .	15
alias2Symbol . . . . .	16
anova.MAList-method . . . . .	19
arrayWeights . . . . .	20
arrayWeightsQuick . . . . .	22
as.data.frame . . . . .	23
as.MAList . . . . .	24
as.matrix . . . . .	24
asMatrixWeights . . . . .	25
auROC . . . . .	26
avearrays . . . . .	27
avedups . . . . .	28
avereps . . . . .	29
backgroundCorrect . . . . .	30
barcodeplot . . . . .	32
beadCountWeights . . . . .	35
blockDiag . . . . .	37
bwss . . . . .	38
bwss.matrix . . . . .	38
camera . . . . .	39
cbind . . . . .	42
changeLog . . . . .	43
classifyTests . . . . .	44
contrastAsCoef . . . . .	45
contrasts.fit . . . . .	47
controlStatus . . . . .	48
coolmap . . . . .	50
cumOverlap . . . . .	51
decideTests . . . . .	52
designI2M . . . . .	54
detectionPValues . . . . .	55
diffSplice . . . . .	56
dim . . . . .	58
dimnames . . . . .	59
dupcor . . . . .	60
eBayes . . . . .	62
EList-class . . . . .	65
exprs.MA . . . . .	66
fitFDist . . . . .	67
fitGammaIntercept . . . . .	69

# Differential gene expression

fitmixture . . . . .	70
fitted.MArrayLM . . . . .	71
genas . . . . .	72
geneSetTest . . . . .	74
getEAWP . . . . .	77
getLayout . . . . .	78
getSpacing . . . . .	79
gls.series . . . . .	80
goana . . . . .	81
gridr . . . . .	86
heatdiagram . . . . .	86
helpMethods . . . . .	88
ids2indices . . . . .	89
imageplot . . . . .	90
imageplot3by2 . . . . .	91
intraspotCorrelation . . . . .	92
is.fullrank . . . . .	93
isNumeric . . . . .	94
kooperberg . . . . .	95
LargeDataObject-class . . . . .	96
limmaUsersGuide . . . . .	97
lm.series . . . . .	98
lmFit . . . . .	99
lmscFit . . . . .	102
loessFit . . . . .	103
logcosh . . . . .	105
logsumexp . . . . .	106
ma3x3 . . . . .	107
makeContrasts . . . . .	108
makeUnique . . . . .	109
MAList-class . . . . .	109
MArrayLM-class . . . . .	110
mdplot . . . . .	111
merge . . . . .	113
mergeScans . . . . .	114
modelMatrix . . . . .	116
modifyWeights . . . . .	117
mrlm . . . . .	118
nec . . . . .	119
normalizeBetweenArrays . . . . .	121
normalizeCyclicLoess . . . . .	123
normalizeForPrintorder . . . . .	125
normalizeMedianAbsValues . . . . .	127
normalizeQuantiles . . . . .	128
normalizeRobustSpline . . . . .	129
normalizeVSN . . . . .	130
normalizeWithinArrays . . . . .	131
normexp.fit . . . . .	133
normexp.fit.control . . . . .	135
normexp.fit.detection.p . . . . .	136
normexp.signal . . . . .	138
plotDensities . . . . .	139

# Differential gene expression

plotExonJunc . . . . .	140
plotExons . . . . .	142
plotFB . . . . .	143
plotlines . . . . .	144
plotMA . . . . .	145
plotMA3by2 . . . . .	147
plotMD . . . . .	148
plotMDS . . . . .	151
plotPrintTipLoess . . . . .	153
plotRLDF . . . . .	154
plotSA . . . . .	156
plotSplice . . . . .	157
plotWithHighlights . . . . .	158
poolVar . . . . .	159
predFCm . . . . .	161
printHead . . . . .	162
PrintLayout . . . . .	163
printorder . . . . .	164
printtipWeights . . . . .	165
propexpr . . . . .	167
propTrueNull . . . . .	168
protectMetachar . . . . .	170
qqt . . . . .	171
QualityWeights . . . . .	172
rankSumTestWithCorrelation . . . . .	173
read.columns . . . . .	174
read.idat . . . . .	176
read.ilmn . . . . .	178
read.ilmn.targets . . . . .	179
read.maimages . . . . .	180
readGAL . . . . .	184
readHeader . . . . .	185
readImaGeneHeader . . . . .	186
readSpotTypes . . . . .	187
readTargets . . . . .	188
removeBatchEffect . . . . .	189
removeExt . . . . .	190
residuals.MArrayLM . . . . .	191
RGList-class . . . . .	192
roast . . . . .	193
romer . . . . .	196
selectModel . . . . .	199
squeezeVar . . . . .	200
strsplit2 . . . . .	202
subsetting . . . . .	203
summary . . . . .	204
targetsA2C . . . . .	205
TestResults-class . . . . .	206
tmixture . . . . .	207
topGO . . . . .	208
topRomer . . . . .	209
topSplice . . . . .	210

# Differential gene expression

toptable . . . . .	211
tricubeMovingAverage . . . . .	214
trigammaInverse . . . . .	215
trimWhiteSpace . . . . .	216
uniquegenelist . . . . .	216
unwrapdups . . . . .	217
venn . . . . .	218
volcanoplot . . . . .	219
voom . . . . .	221
vooma . . . . .	223
voomWithQualityWeights . . . . .	224
weighted.median . . . . .	226
weightedLoess . . . . .	227
write.fit . . . . .	228
wsva . . . . .	229
zscore . . . . .	230

# Differential gene expression

---

lmFit

*Linear Model for Series of Arrays*

---

## Description

Fit linear model for each gene given a series of arrays

## Usage

```
lmFit(object, design=NULL, ndups=1, spacing=1, block=NULL, correlation, weights=NULL  
      method="ls", ...)
```

## Arguments

object	A matrix-like data object containing log-ratios or log-expression values for a series of arrays, with rows corresponding to genes and columns to samples. Any type of data object that can be processed by <a href="#">getEAWP</a> is acceptable.
design	the design matrix of the microarray experiment, with rows corresponding to arrays and columns to coefficients to be estimated. Defaults to the unit vector meaning that the arrays are treated as replicates.
ndups	positive integer giving the number of times each distinct probe is printed on each array.
spacing	positive integer giving the spacing between duplicate occurrences of the same probe, spacing=1 for consecutive rows.
block	vector or factor specifying a blocking variable on the arrays. Has length equal to the number of arrays. Must be NULL if ndups>2.
correlation	the inter-duplicate or inter-technical replicate correlation
weights	non-negative precision weights. Can be a numeric matrix of individual weights of same size as the object expression matrix, or a numeric vector of array weights with length equal to ncol of the expression matrix, or a numeric vector of gene weights with length equal to nrow of the expression matrix.
method	fitting method; "ls" for least squares or "robust" for robust regression
...	other optional arguments to be passed to lm.series, gls.series or mrlm

# Differential gene expression

## Details

This function fits multiple linear models by weighted or generalized least squares. It accepts data from a experiment involving a series of microarrays with the same set of probes. A linear model is fitted to the expression data for each probe. The expression data should be log-ratios for two-color array platforms or log-expression values for one-channel platforms. (To fit linear models to the individual channels of two-color array data, see [lmscFit](#).) The coefficients of the fitted models describe the differences between the RNA sources hybridized to the arrays. The probe-wise fitted model results are stored in a compact form suitable for further processing by other functions in the limma package.

# Differential gene expression

The function allows for missing values and accepts quantitative precision weights through the weights argument. It also supports two different correlation structures. If block is not NULL then different arrays are assumed to be correlated. If block is NULL and ndups is greater than one then replicate spots on the same array are assumed to be correlated. It is not possible at this time to fit models with both a block structure and a duplicate-spot correlation structure simultaneously.

# Differential gene expression

If object is a matrix then it should contain log-ratios or log-expression data with rows corresponding to probes and columns to arrays. (A numeric vector is treated the same as a matrix with one column.) For objects of other classes, a matrix of expression values is taken from the appropriate component or slot of the object. If object is of class `MAList` or `marrayNorm`, then the matrix of log-ratios (M-values) is extracted. If object is of class `ExpressionSet`, then the expression matrix is extracted. (This may contain log-expression or log-ratio values, depending on the platform.) If object is of class `PLMset` then the matrix of chip coefficients `chip.coefs` is extracted.

# Differential gene expression

The correlation argument has a default value of `0.75`, but in normal use this default value should not be relied on and the correlation value should be estimated using the function `duplicateCorrelation`. The default value is likely to be too high in particular if used with the `block` argument.

The actual linear model computations are done by passing the data to one the lower-level functions `lm.series`, `gls.series` or `mrlm`. The function `mrlm` is used if `method="robust"`. If `method="ls"`, then `gls.series` is used if a correlation structure has been specified, i.e., if `ndups>1` or `block` is non-null and `correlation` is different from zero. If `method="ls"` and there is no correlation structure, `lm.series` is used.

# Differential gene expression

```
genes # Installing Limma package
```

```
# if (!requireNamespace("BiocManager", quietly = TRUE))
#   install.packages("BiocManager")
# BiocManager::install("limma", version = "3.8")
```

```
library(limma)
```

```
##
## Attaching package: 'limma'
```

```
## The following object is masked from 'package:BiocGenerics':
##
##     plotMA
```

```
genotype<- factor(pData(TRGN599_data)[,2] , levels = levels(pData(TRGN599_data)[,2]))
stimul <- factor(pData(TRGN599_data)[,3] , levels = levels(pData(TRGN599_data)[,3]))
TRGN599_design<- model.matrix(~genotype)
```

```
TRGN599_fit <- lmFit(TRGN599_matexp,TRGN599_design)
TRGN599_fit
```

# Differential gene expression

```
## An object of class "MArrayLM"
## $coefficients
##                               (Intercept)  genotypeWT →Represents the difference
## 1415670_PM_at             8.280645  0.09075880 in the predicted value of Y
## 1415671_PM_at             9.256379 -0.64344298 for each one-unit
## 1415672_PM_at             9.596247 -0.18578687 difference in  $X_1$ , if
## 1415673_PM_at             7.793342 -0.89930999  $X_2$  remains constant.
## 1415674_PM_a_at           8.417905  0.07286682
## 45136 more rows ...
##
## $rank
## [1] 2
##
## $assign
## [1] 0 1
```

The intercept, can be interpreted as the value you would predict for Y if both  $X_1 = 0$  and  $X_2 = 0$ .

# Differential gene expression

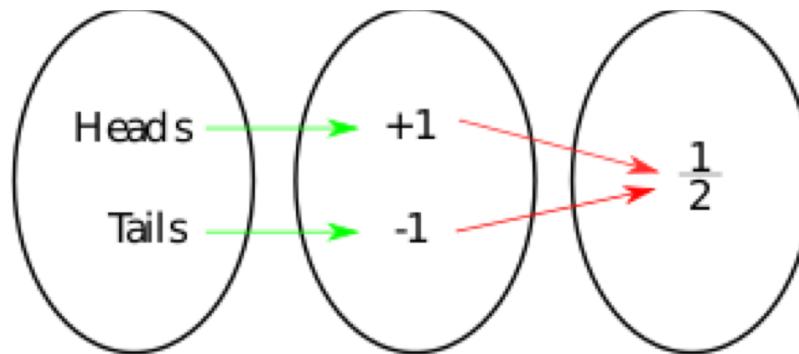
```
## $qr
##   (Intercept) genotypeWT
## 1 -3.4641016 -1.7320508
## 2  0.2886751 -1.7320508
## 3  0.2886751  0.2240092
## 4  0.2886751  0.2240092
## 5  0.2886751  0.2240092
## 7 more rows ...
##
## $qraux
## [1] 1.288675 1.224009
##
## $pivot
## [1] 1 2
##
## $tol
## [1] 1e-07
##
## $rank
## [1] 2
```

# Differential gene expression

```
## $df.residual
## [1] 10 10 10 10 10
## 45136 more elements ...
##
## $sigma
##   1415670_PM_at   1415671_PM_at   1415672_PM_at   1415673_PM_at
##   0.2384200       0.2007284       0.1439598       0.7019110
## 1415674_PM_a_at
##   0.2357675
## 45136 more elements ...
##
## $cov.coefficients
##             (Intercept) genotypeWT
## (Intercept)  0.1666667 -0.1666667
## genotypeWT -0.1666667  0.3333333
```

# Differential gene expression

- Covariance:
  - Measure of the joint variability of two random variables.



- The covariance between two jointly distributed real-valued random variables is defined as the expected product of their deviations from their individual expected values

$$\text{cov}(X, Y) = E [(X - E[X])(Y - E[Y])], \quad (\text{Eq.1})$$

# Differential gene expression

```
## $stdev.unscaled
##                               (Intercept) genotypeWT
## 1415670_PM_at      0.4082483  0.5773503
## 1415671_PM_at      0.4082483  0.5773503
## 1415672_PM_at      0.4082483  0.5773503
## 1415673_PM_at      0.4082483  0.5773503
## 1415674_PM_a_at    0.4082483  0.5773503
## 45136 more rows ...
##
## $pivot
## [1] 1 2
##
## $Amean
##   1415670_PM_at  1415671_PM_at  1415672_PM_at  1415673_PM_at
##   8.326024        8.934657        9.503353        7.343687
## 1415674_PM_a_at
##   8.454338
## 45136 more elements ...
##
## $method
## [1] "ls"
##
## $design
##   (Intercept) genotypeWT
## 1           1           1
## 2           1           1
## 3           1           1
## 4           1           1
## 5           1           1
## 7 more rows ...
```

# Differential gene expression

```
TRGN599_fit<-eBayes(TRGN599_fit)
```

```
TRGN599_dg_top_50 <- topTable(TRGN599_fit, coef = 2, adjust = "fdr", n = 50)
```

# Differential gene expression

- Bayes' theorem
  - Alternatively Bayes' law or Bayes' rule
  - Describes the probability of an event, based on prior knowledge of conditions that might be related to the event.
  - Is stated mathematically as the following equation:

$$P(A | B) = \frac{P(B | A) P(A)}{P(B)}$$

# Differential gene expression

- Top 50 probes of the differential gene expression:
  - Table are obtained based on the adjusted p-values of the moderated t-test that is used here to estimate the statistical significance of the differences.

```
179
180 ~ ````{R}
181 TRGN599_dg_top_50 <- topTable(TRGN599_fit, coef = 2, adjust = "fdr", n = 50)
182 head(TRGN599_dg_top_50)
183 ````
```

	logFC	AveExpr	t	P.Value	adj.P.Val	B
1419758_PM_at	-3.972031	6.200538	-30.06466	7.699681e-13	3.475713e-08	14.348063
1419759_PM_at	-3.648735	6.532582	-24.20075	1.050411e-11	2.370831e-07	13.384220
1434909_PM_at	-2.955792	5.339058	-19.72618	1.208176e-10	1.817943e-06	12.226986
1434437_PM_x_at	-1.656676	5.756127	-18.69385	2.286562e-10	2.580442e-06	11.881989
1449874_PM_at	-1.576391	4.107422	-15.16105	2.685119e-09	2.424179e-05	10.384427
1448226_PM_at	-1.624406	4.320060	-13.04231	1.531973e-08	1.152580e-04	9.173986

# Differential gene expression

- Top 50 probes of the differential gene expression:
  - Interpreting the columns:
    - AveExpr – Average log2-expression for the probe over all arrays and channels.
      - It is an indicator of the general expression level of the gene.
    - LogFC – The log2-fold-change corresponding to the investigated factor (genotype of the samples in the example).
      - It shows how much different the expression is between the two compared categories.
      - Positive values mean over-expression
      - Negative values means repression.
      - Since it is the 2-based logarithm of the fold changes, value one can be interpreted as a two fold up-regulation.

# Differential gene expression

- Top 50 probes of the differential gene expression:
  - Interpreting the columns:
    - t, P.Value, adj.P.Val – The outputs of the moderated t-test.
    - The raw p-value is the false discovery rate.
    - The adjusted p-value reveals the significance level of the difference.
  - B – Log-odds that the gene expression is different between the compared categories.
    - This measure describes how likely that the differential expression is real.
    - The higher the score, the more likely is that the results are meaningful.
    - Usually, the table is sorted by this B log-odds score.

# Differential gene expression

- Let us compare the stimulated and the unstimulated samples next:

```
185 # Comparing stimulating vs unstimulating samples
186 ````{R}
187 TRGN599_design.1 <- model.matrix(~stimul)
188 TRGN599_fit.1 <- lmFit(TRGN599_matexp, TRGN599_design.1)
189 TRGN599_fit.1 <- eBayes(TRGN599_fit.1)
190 TRGN599_dg.top.50.stim <- topTable(TRGN599_fit.1, coef = 2, adjust = "fdr", n=50)
191 head(TRGN599_dg.top.50.stim)
192 ````
```

	logFC	AveExpr	t	P.Value	adj.P.Val	B
1418322_PM_at	5.451960	6.313120	62.71547	4.793074e-19	2.163642e-14	31.63014
1425822_PM_a_at	-4.981851	7.496269	-44.71355	6.278585e-17	1.417108e-12	28.21168
1449037_PM_at	6.061854	6.871325	42.99404	1.103433e-16	1.660336e-12	27.76799
1417038_PM_at	-3.884873	8.236228	-40.30581	2.790146e-16	3.148749e-12	27.01878
1454893_PM_at	-2.367691	8.834408	-39.51840	3.703832e-16	3.343894e-12	26.78542
1452621_PM_at	-2.834306	7.321527	-38.26152	5.889866e-16	4.431241e-12	26.39887

# Annotating Data

## Annotating Data

**Comment:** It can be used the `mouse.annot <- read.delim("GPL11180-26917.txt",skip=16,row.names=1)`

```
TRGN599_mouse_annot <- read.table('Users/enriquevelazquez/Documents/R_working_directory/TRGN599_CelFiles/TRGN599_selected_annotation.txt')
TRGN599_sel_annot <- TRGN599_mouse_annot[row.names(TRGN599_dg_top_50),]

TRGN599_dg_top_50_annot <- cbind(TRGN599_dg_top_50,TRGN599_sel_annot)
```

## Generating a Table

```
write.table(TRGN599_dg_top_50_annot[order(TRGN599_dg_top_50_annot$logFC),], "TRGN599_diff_genes.txt")
```

## Adjusting Pvalues

```
TRGN599_selected <- p.adjust(TRGN599_fit$p.value[, 2],method="fdr") <0.03
```

## Generating an expression Matrix

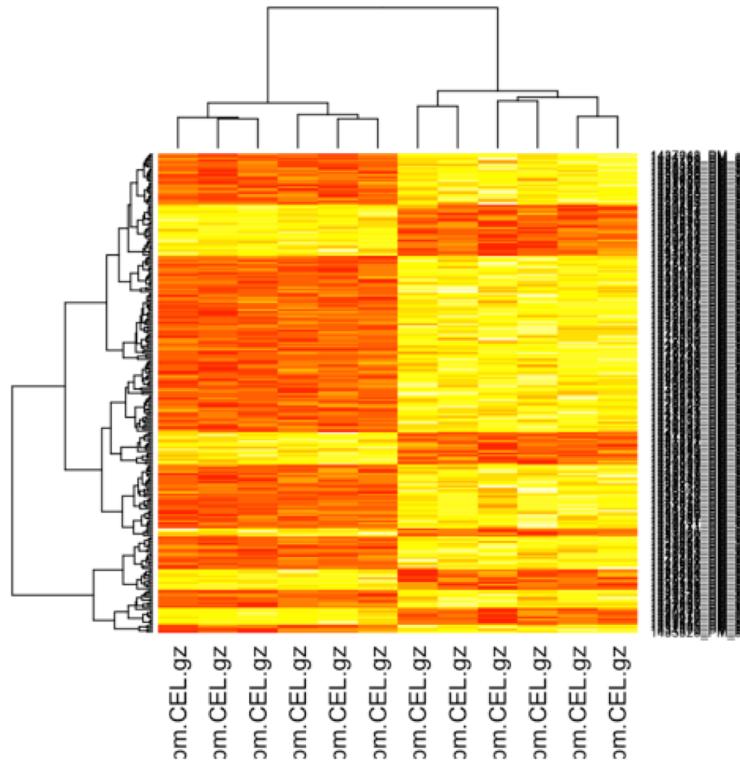
```
TRGN599_matexp_gen <- TRGN599_matexp[TRGN599_selected,]
dim(TRGN599_matexp_gen)

## [1] 234 12
```

# Annotating Data

## Generating a heatmap

```
heatmap(TRGN599_matexp_gen)
```



## Generating a Model-design

```
TRGN599_design_1 <- model.matrix(~stimul)
TRGN599_fit_1 <- lmFit(TRGN599_matexp,TRGN599_design_1)
TRGN599_fit_1<-eBayes(TRGN599_fit_1)
```

```
TRGN599_dg_top_50_stim <- topTable(TRGN599_fit_1, coef = 2, adjust = "fdr", n = 50)
TRGN599_sel_annotation <- TRGN599_mouse_annotation[row.names(TRGN599_dg_top_50_stim),]
TRGN599_dg_top_50_stim_annotation <- cbind(TRGN599_dg_top_50_stim,TRGN599_sel_annotation)
```

# Annotating Data

## Generating a Model-design

```
TRGN599_design_1 <- model.matrix(~stimul)
TRGN599_fit_1 <- lmFit(TRGN599_matexp,TRGN599_design_1)
TRGN599_fit_1<-eBayes(TRGN599_fit_1)
```

```
TRGN599_dg_top_50_stim <- topTable(TRGN599_fit_1, coef = 2, adjust = "fdr", n = 50)
TRGN599_sel_annot <- TRGN599_mouse_annot[row.names(TRGN599_dg_top_50_stim),]
TRGN599_dg_top_50_stim_annot <- cbind(TRGN599_dg_top_50_stim,TRGN599_sel_annot)
```

## Generating a table with diff genes-stimulation

```
write.table(TRGN599_dg_top_50_stim_annot[order(TRGN599_dg_top_50_stim_annot$logFC),],"TRGN599_diffgenes_stimulation.txt")
```

## Annotating first created table using Gene symbols

```
TRGN599_dg_top_50_annot$Gene.Symbol
```

```
## [1] Abcb1a          Abcb1a          Rragd
## [4] Rrm2            Ly96            Rrm2
## [7] Ripk3           Acot7           2010002M12Rik
## [10] Tsc1            Top2a           Ada
## [13] Gpr137b-ps     Ccna2           Lpin2
## [16] Mki67           Gtf2f1           Atp6ap2
## [19] Mfsd1           Pcsk1           Gpraspl
## [22] Lpin2           Atp6ap2          Lpxn
## [25] Gtf2ird1        Gpr18            Npc2
## [28] Scarb2          Uhrf1           4833442J19Rik
## [31] Idil             Abcb1b           Igj
## [34] Atpif1          Crim1           Fam129a
## [37] Tpp1             Fam26f           Ttf1
## [40] Fam125a         Ttc28            Incenp
## [43] Scpep1          Gale             Serpinb6b
## [46] Ncapg            Gpr137b          Cxcr3
## [49] Gm4204 /// Nap111 Hsd17b11
## 21762 Levels: 0610005C13Rik 0610006L08Rik 0610007C21Rik ... Zzz3
```

# Accessing data from CEL files

## Annotating second generated table - stimulation - using Gene symbols

```
TRGN599_dg_top_50_stim_annot$Gene.Symbol
```

```
## [1] Crem          Dtx1           Crem
## [4] Sept9         Fam189b        Pcbd2
## [7] Rab37         Tlr1           Trpv2
## [10] Hsd11b1       Mgst2          Emp3
## [13] Dap11         Zfp295         Rnf144a
## [16] Adcy7          1110054M08Rik Fosl2 /// LOC634417
## [19] Slco3a1       Cisd3          Tmem55a
## [22] Taf9b         Nr4a3          Dhrs7
## [25] Prps2         Rab37          Cryl1
## [28] Zfp52         Arsb           Pus3
## [31] Dbp            I12            Zeb1
## [34] Cdca71        Adcy7          Irf4
## [37] Iqgap2        Hspa4l         Mirlet7d
## [40] Rgs14          Fermt3         Tet1
## [43] Pycard         Zfp52          Afp
## [46] Gchl           Jakmip1        Dci
## [49] Naglu          Rcsd1
## 21762 Levels: 0610005C13Rik 0610006L08Rik 0610007C21Rik ... Zzz3
```