RNA-seq edgeR limma

LAC

2022-10-13

RNA-seq data analysis workflow using **edgeR** package to import, organize, filter and normalize data, followed by the **limma** package with the *voom* method, linear modeling and empirical Bayes moderation to assess differential expression. The **Glimma** package can be used in addition to enable interactive exploration of the results at both sample and gene-level.

1. Set-up

```
library("limma")
library("Glimma")
library("edgeR")
library("Homo.sapiens")
#library("DEFormats")

library("airway")
library("gplots")
library("RColorBrewer")
```

2. Data loading and organization

Data used in this workflow come from the Bioconductor *airway* package that summarizes an RNA-seq experiment (i.e. a *RangedSummarizedExperiment* object of RNA-seq gene read counts) wherein each of 4 primary human airway smooth muscle (ASM) cell lines were treated with dexamethasone (1 μM; 18 hrs) - a synthetic potent glucocorticoid steroid with anti-inflammatory effects used as medication for individuals with asthma to reduce the inflammation of the airways, or left untreated.

GEO GSE52778 Illumina HiSeq 2000 (Homo sapiens) platform

```
data(airway)
airway
```

```
## class: RangedSummarizedExperiment
## dim: 64102 8
## metadata(1): ''
## assays(1): counts
## rownames(64102): ENSG00000000003 ENSG00000000005 ... LRG_98 LRG_99
## rowData names(0):
## colnames(8): SRR1039508 SRR1039509 ... SRR1039520 SRR1039521
## colData names(9): SampleName cell ... Sample BioSample
```

The **assay** component of the *RangedSummarizedExperiment* object contains the count matrix.

- **rownames** - genes
- **colnames** - samples

The **colData** contains sample information.

```
head(assay(airway))
```

```
##                 SRR1039508 SRR1039509 SRR1039512 SRR1039513 SRR1039516
## ENSG00000000003        679        448        873        408       1138
## ENSG00000000005          0          0          0          0          0
## ENSG00000000419        467        515        621        365        587
## ENSG00000000457        260        211        263        164        245
## ENSG00000000460         60         55         40         35         78
## ENSG00000000938          0          0          2          0          1
##                 SRR1039517 SRR1039520 SRR1039521
## ENSG00000000003       1047        770        572
## ENSG00000000005          0          0          0
## ENSG00000000419        799        417        508
## ENSG00000000457        331        233        229
## ENSG00000000460         63         76         60
## ENSG00000000938          0          0          0
```

```
as.data.frame(colData(airway))
```

```
##            SampleName    cell   dex albut        Run avgLength Experiment
## SRR1039508 GSM1275862  N61311 untrt untrt SRR1039508       126  SRX384345
## SRR1039509 GSM1275863  N61311   trt untrt SRR1039509       126  SRX384346
## SRR1039512 GSM1275866 N052611 untrt untrt SRR1039512       126  SRX384349
## SRR1039513 GSM1275867 N052611   trt untrt SRR1039513        87  SRX384350
## SRR1039516 GSM1275870 N080611 untrt untrt SRR1039516       120  SRX384353
## SRR1039517 GSM1275871 N080611   trt untrt SRR1039517       126  SRX384354
## SRR1039520 GSM1275874 N061011 untrt untrt SRR1039520       101  SRX384357
## SRR1039521 GSM1275875 N061011   trt untrt SRR1039521        98  SRX384358
##               Sample    BioSample
## SRR1039508 SRS508568 SAMN02422669
## SRR1039509 SRS508567 SAMN02422675
## SRR1039512 SRS508571 SAMN02422678
## SRR1039513 SRS508572 SAMN02422670
## SRR1039516 SRS508575 SAMN02422682
## SRR1039517 SRS508576 SAMN02422673
## SRR1039520 SRS508579 SAMN02422683
## SRR1039521 SRS508580 SAMN02422677
```

2.1 Convert the RangedSummarizedExperiment object to DGEList object

Using the **edgeR::SE2DGEList** function, the counts of the assay component of the input RangedSummarizedExperiment data object is extracted and used as the counts component of the output DGEList object.

The rowRanges or rowData of the input is converted into a data.frame and used as genes in the output.

The colData of the input is also converted into a data.frame and used as the sample information in the output.

DGEList object holds the dataset to be analysed by edgeR and the subsequent calculations performed on the dataset.

Specifically it contains:

column	description
counts	numeric matrix containing the read counts
lib.size	numeric vector containing the total count (seq depth) to normalize against for each sample
norm.factors	numeric vector containing normalization factors that modify library sizes
group	vector giving the experimental group/condition for each sample/library

```
dge <- edgeR::SE2DGEList(airway)
dim(dge)
```

```
## [1] 64102     8
```

```
dge$samples
```

```
##            group lib.size norm.factors SampleName    cell   dex albut
## SRR1039508     1 20637971            1 GSM1275862  N61311 untrt untrt
## SRR1039509     1 18809481            1 GSM1275863  N61311   trt untrt
## SRR1039512     1 25348649            1 GSM1275866 N052611 untrt untrt
## SRR1039513     1 15163415            1 GSM1275867 N052611   trt untrt
## SRR1039516     1 24448408            1 GSM1275870 N080611 untrt untrt
## SRR1039517     1 30818215            1 GSM1275871 N080611   trt untrt
## SRR1039520     1 19126151            1 GSM1275874 N061011 untrt untrt
## SRR1039521     1 21164133            1 GSM1275875 N061011   trt untrt
##                  Run avgLength Experiment    Sample    BioSample
## SRR1039508 SRR1039508       126  SRX384345 SRS508568 SAMN02422669
## SRR1039509 SRR1039509       126  SRX384346 SRS508567 SAMN02422675
## SRR1039512 SRR1039512       126  SRX384349 SRS508571 SAMN02422678
## SRR1039513 SRR1039513        87  SRX384350 SRS508572 SAMN02422670
## SRR1039516 SRR1039516       120  SRX384353 SRS508575 SAMN02422682
## SRR1039517 SRR1039517       126  SRX384354 SRS508576 SAMN02422673
## SRR1039520 SRR1039520       101  SRX384357 SRS508579 SAMN02422683
## SRR1039521 SRR1039521        98  SRX384358 SRS508580 SAMN02422677
```

```
head(rownames(dge)) # genes
```

```
## [1] "ENSG00000000003" "ENSG00000000005" "ENSG00000000419" "ENSG00000000457"
## [5] "ENSG00000000460" "ENSG00000000938"
```

```
head(dge$counts)
```

```
##                 SRR1039508 SRR1039509 SRR1039512 SRR1039513 SRR1039516
## ENSG00000000003        679        448        873        408       1138
## ENSG00000000005          0          0          0          0          0
## ENSG00000000419        467        515        621        365        587
## ENSG00000000457        260        211        263        164        245
## ENSG00000000460         60         55         40         35         78
## ENSG00000000938          0          0          2          0          1
##                 SRR1039517 SRR1039520 SRR1039521
## ENSG00000000003       1047        770        572
## ENSG00000000005          0          0          0
## ENSG00000000419        799        417        508
## ENSG00000000457        331        233        229
## ENSG00000000460         63         76         60
## ENSG00000000938          0          0          0
```

2.2 Assign the experimental group/condition for each sample

```
dge$samples$group <- dge$samples$dex
dge$samples
```

```
##            group lib.size norm.factors SampleName    cell   dex albut
## SRR1039508 untrt 20637971            1 GSM1275862  N61311 untrt untrt
## SRR1039509   trt 18809481            1 GSM1275863  N61311   trt untrt
## SRR1039512 untrt 25348649            1 GSM1275866 N052611 untrt untrt
## SRR1039513   trt 15163415            1 GSM1275867 N052611   trt untrt
## SRR1039516 untrt 24448408            1 GSM1275870 N080611 untrt untrt
## SRR1039517   trt 30818215            1 GSM1275871 N080611   trt untrt
## SRR1039520 untrt 19126151            1 GSM1275874 N061011 untrt untrt
## SRR1039521   trt 21164133            1 GSM1275875 N061011   trt untrt
##                     Run avgLength Experiment    Sample    BioSample
## SRR1039508 SRR1039508       126  SRX384345 SRS508568 SAMN02422669
## SRR1039509 SRR1039509       126  SRX384346 SRS508567 SAMN02422675
## SRR1039512 SRR1039512       126  SRX384349 SRS508571 SAMN02422678
## SRR1039513 SRR1039513        87  SRX384350 SRS508572 SAMN02422670
## SRR1039516 SRR1039516       120  SRX384353 SRS508575 SAMN02422682
## SRR1039517 SRR1039517       126  SRX384354 SRS508576 SAMN02422673
## SRR1039520 SRR1039520       101  SRX384357 SRS508579 SAMN02422683
## SRR1039521 SRR1039521        98  SRX384358 SRS508580 SAMN02422677
```

2.3 Annotate the genes: ENSEMBL to ENTREZ using the Homo.sapiens library

```
geneid <- rownames(dge)
genes <- select(Homo.sapiens, keys=geneid, columns=c("ENTREZID", "SYMBOL", "TXCHROM"),
                keytype="ENSEMBL")
head(genes)
```

```
##           ENSEMBL ENTREZID   SYMBOL TXCHROM
## 1 ENSG00000000003     7105   TSPAN6    chrX
## 2 ENSG00000000005    64102     TNMD    chrX
## 3 ENSG00000000419     8813     DPM1   chr20
## 4 ENSG00000000457    57147    SCYL3    chr1
## 5 ENSG00000000460    55732 C1orf112    chr1
## 6 ENSG00000000938     2268      FGR    chr1
```

```
dim(genes)
```

```
## [1] 71739     4
```

Remove duplicated gene IDs

```
genes <- genes[!duplicated(genes$ENSEMBL),]
dim(genes)
```

```
## [1] 64102     4
```

2.4 Package the DGEList object to contain raw count data, associated sample information and gene annotations

```
dge$genes <- genes
dge
```

```
## An object of class "DGEList"
## $counts
##                 SRR1039508 SRR1039509 SRR1039512 SRR1039513 SRR1039516
## ENSG00000000003        679        448        873        408       1138
## ENSG00000000005          0          0          0          0          0
## ENSG00000000419        467        515        621        365        587
## ENSG00000000457        260        211        263        164        245
## ENSG00000000460         60         55         40         35         78
##                 SRR1039517 SRR1039520 SRR1039521
## ENSG00000000003       1047        770        572
## ENSG00000000005          0          0          0
## ENSG00000000419        799        417        508
## ENSG00000000457        331        233        229
## ENSG00000000460         63         76         60
## 64097 more rows ...
##
## $samples
##            group lib.size norm.factors SampleName    cell   dex albut
## SRR1039508 untrt 20637971            1 GSM1275862  N61311 untrt untrt
## SRR1039509   trt 18809481            1 GSM1275863  N61311   trt untrt
## SRR1039512 untrt 25348649            1 GSM1275866 N052611 untrt untrt
## SRR1039513   trt 15163415            1 GSM1275867 N052611   trt untrt
## SRR1039516 untrt 24448408            1 GSM1275870 N080611 untrt untrt
## SRR1039517   trt 30818215            1 GSM1275871 N080611   trt untrt
## SRR1039520 untrt 19126151            1 GSM1275874 N061011 untrt untrt
```

```
## SRR1039521   trt 21164133           1 GSM1275875 N061011   trt untrt
##                   Run avgLength Experiment    Sample    BioSample
## SRR1039508 SRR1039508       126  SRX384345 SRS508568 SAMN02422669
## SRR1039509 SRR1039509       126  SRX384346 SRS508567 SAMN02422675
## SRR1039512 SRR1039512       126  SRX384349 SRS508571 SAMN02422678
## SRR1039513 SRR1039513        87  SRX384350 SRS508572 SAMN02422670
## SRR1039516 SRR1039516       120  SRX384353 SRS508575 SAMN02422682
## SRR1039517 SRR1039517       126  SRX384354 SRS508576 SAMN02422673
## SRR1039520 SRR1039520       101  SRX384357 SRS508579 SAMN02422683
## SRR1039521 SRR1039521        98  SRX384358 SRS508580 SAMN02422677
##
## $genes
##           ENSEMBL ENTREZID   SYMBOL TXCHROM
## 1 ENSG00000000003     7105   TSPAN6    chrX
## 2 ENSG00000000005    64102     TNMD    chrX
## 3 ENSG00000000419     8813     DPM1   chr20
## 4 ENSG00000000457    57147    SCYL3    chr1
## 5 ENSG00000000460    55732 C1orf112    chr1
## 64097 more rows ...
```

3. Data pre-processing

3.1 CPM (counts per million) and log-CPM transformations from raw-scale

Transform raw counts onto a scale that accounts for library size differences using the **cpm** function from edgeR. RPKM or FPKM transformations can also be used to account for gene length differences. Use CPM as here the differential expression analysis looks for gene expression changes between conditions, and not changes in gene lengths that may be of interest in comparing expression across multiple genes, for instance.

```
avg_lib_size <- mean(dge$samples$lib.size) * 1e-6 #million
avg_lib_size
```

```
## [1] 21.93955
```

```
median_lib_size <- median(dge$samples$lib.size) * 1e-6 #million
median_lib_size
```

```
## [1] 20.90105
```

```
cpm <- edgeR::cpm(dge)
lcpm <- edgeR::cpm(dge, log = TRUE)
summary(lcpm)
```

```
##    SRR1039508        SRR1039509        SRR1039512        SRR1039513
##  Min.   :-3.4555   Min.   :-3.4555   Min.   :-3.4555   Min.   :-3.455
##  1st Qu.:-3.4555   1st Qu.:-3.4555   1st Qu.:-3.4555   1st Qu.:-3.455
##  Median :-3.4555   Median :-3.4555   Median :-3.4555   Median :-3.455
##  Mean   :-1.5005   Mean   :-1.5177   Mean   :-1.5077   Mean   :-1.559
##  3rd Qu.:-0.7966   3rd Qu.:-0.9532   3rd Qu.:-0.8248   3rd Qu.:-1.038
##  Max.   :13.8173   Max.   :13.7305   Max.   :14.3069   Max.   :14.141
##    SRR1039516        SRR1039517        SRR1039520        SRR1039521
```

```
##  Min.   :-3.4555   Min.   :-3.455   Min.   :-3.4555   Min.   :-3.455
##  1st Qu.:-3.4555   1st Qu.:-3.455   1st Qu.:-3.4555   1st Qu.:-3.455
##  Median :-3.4555   Median :-3.455   Median :-3.4555   Median :-3.455
##  Mean   :-1.5069   Mean   :-1.535   Mean   :-1.4999   Mean   :-1.552
##  3rd Qu.:-0.8861   3rd Qu.:-1.057   3rd Qu.:-0.8321   3rd Qu.:-1.092
##  Max.   :13.9900   Max.   :13.669   Max.   :14.2737   Max.   :14.103
```

3.2 Filter lowly expressed genes

filterByExpr function determines which genes have sufficiently large counts to be retained in a statistical analysis.

The function keeps genes with about 10 read counts or more in a minimum number of samples, where the number of samples is chosen according to the minimum group sample size.

```
# number of genes unexpressed across all 8 sample
table(rowSums(dge$counts==0)==8)
```

```
##
## FALSE  TRUE
## 33469 30633
```

```
# automated gene filtering with filterByExpr keeping as many genes as possible with worthwhile counts
keep.exprs <- edgeR::filterByExpr(dge, group=dge$samples$group)
dge <- dge[keep.exprs,,keep.lib.sizes=FALSE]
dim(dge)
```

```
## [1] 15926     8
```

```
dge
```

```
## An object of class "DGEList"
## $counts
##                 SRR1039508 SRR1039509 SRR1039512 SRR1039513 SRR1039516
## ENSG00000000003        679        448        873        408       1138
## ENSG00000000419        467        515        621        365        587
## ENSG00000000457        260        211        263        164        245
## ENSG00000000460         60         55         40         35         78
## ENSG00000000971       3251       3679       6177       4252       6721
##                 SRR1039517 SRR1039520 SRR1039521
## ENSG00000000003       1047        770        572
## ENSG00000000419        799        417        508
## ENSG00000000457        331        233        229
## ENSG00000000460         63         76         60
## ENSG00000000971      11027       5176       7995
## 15921 more rows ...
##
## $samples
##             group lib.size norm.factors SampleName    cell   dex albut
## SRR1039508 untrt 20608402            1 GSM1275862  N61311 untrt untrt
## SRR1039509   trt 18783120            1 GSM1275863  N61311   trt untrt
## SRR1039512 untrt 25311320            1 GSM1275866 N052611 untrt untrt
```

```
## SRR1039513   trt 15144524            1 GSM1275867 N052611   trt untrt
## SRR1039516 untrt 24411867            1 GSM1275870 N080611 untrt untrt
## SRR1039517   trt 30776089            1 GSM1275871 N080611   trt untrt
## SRR1039520 untrt 19094104            1 GSM1275874 N061011 untrt untrt
## SRR1039521   trt 21135511            1 GSM1275875 N061011   trt untrt
##                   Run avgLength Experiment    Sample    BioSample
## SRR1039508 SRR1039508       126  SRX384345 SRS508568 SAMN02422669
## SRR1039509 SRR1039509       126  SRX384346 SRS508567 SAMN02422675
## SRR1039512 SRR1039512       126  SRX384349 SRS508571 SAMN02422678
## SRR1039513 SRR1039513        87  SRX384350 SRS508572 SAMN02422670
## SRR1039516 SRR1039516       120  SRX384353 SRS508575 SAMN02422682
## SRR1039517 SRR1039517       126  SRX384354 SRS508576 SAMN02422673
## SRR1039520 SRR1039520       101  SRX384357 SRS508579 SAMN02422683
## SRR1039521 SRR1039521        98  SRX384358 SRS508580 SAMN02422677
##
## $genes
##           ENSEMBL ENTREZID   SYMBOL TXCHROM
## 1 ENSG00000000003     7105   TSPAN6    chrX
## 3 ENSG00000000419     8813     DPM1   chr20
## 4 ENSG00000000457    57147    SCYL3    chr1
## 5 ENSG00000000460    55732 C1orf112    chr1
## 7 ENSG00000000971     3075      CFH    chr1
## 15921 more rows ...
```

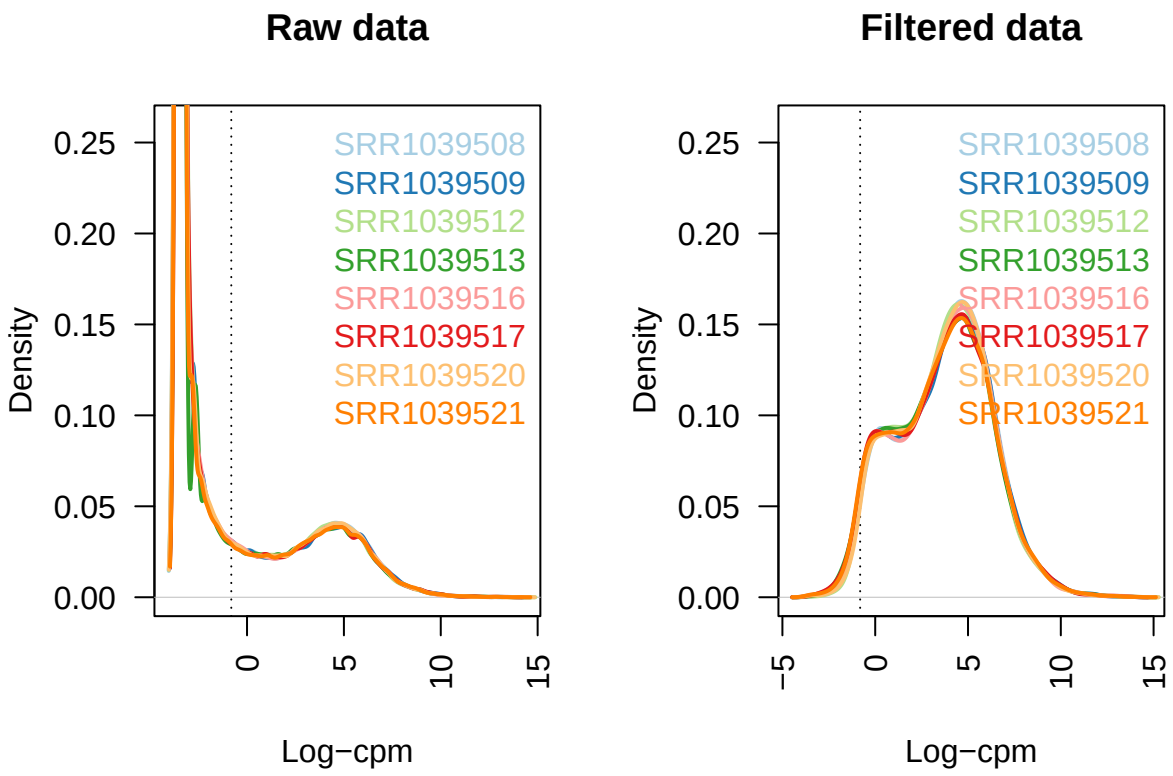
```
lcpm.cutoff <- log2(10/median_lib_size + 2/avg_lib_size)

samplenames <- colnames(dge)
nsamples <- ncol(dge)
col <- brewer.pal(nsamples, "Paired")
par(mfrow=c(1,2))
plot(density(lcpm[,1]), col=col[1], lwd=2, ylim=c(0,0.26), las=2, main="", xlab="")
title(main="Raw data", xlab="Log-cpm")
abline(v=lcpm.cutoff, lty=3)

for (i in 2:nsamples){
  den <- density(lcpm[,i])
  lines(den$x, den$y, col=col[i], lwd=2)
}
legend("topright", samplenames, text.col=col, bty="n")

lcpm <- cpm(dge, log=TRUE)
plot(density(lcpm[,1]), col=col[1], lwd=2, ylim=c(0,0.26), las=2, main="", xlab="")
title(main="Filtered data", xlab="Log-cpm")
abline(v=lcpm.cutoff, lty=3)

for (i in 2:nsamples){
  den <- density(lcpm[,i])
  lines(den$x, den$y, col=col[i], lwd=2)
}
legend("topright", samplenames, text.col=col, bty="n")
```



3.3 Normalizing gene expression distributions

Calculate scaling factors to convert raw library sizes into effective library sizes with the **calcNormFactors** using the TMM method implements the trimmed mean of M-values (TMM) method.

The effective library sizes for use in downstream analysis are `lib.size` $*$ `norm.factors` where lib.size contains the original library sizes and norm.factors is the vector of scaling factors computed by this function.

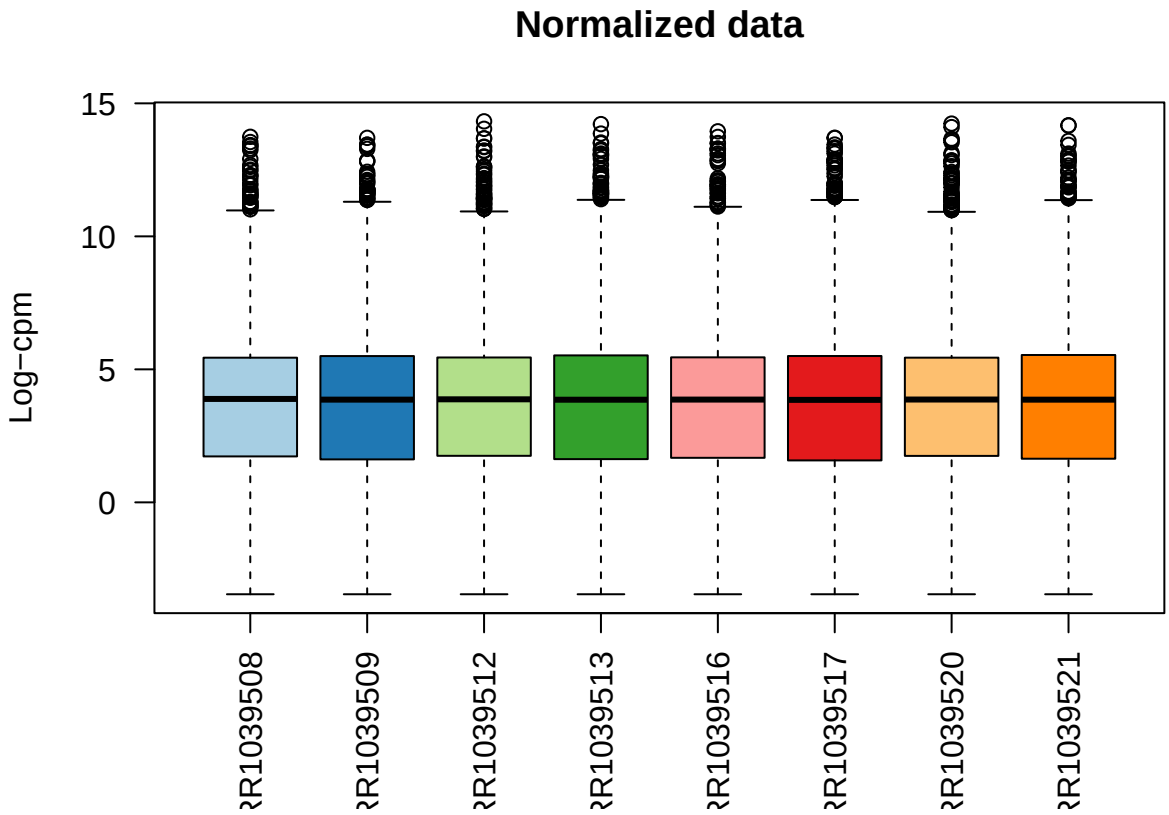
The calculated normalization factors, `norm.factors` are used as scaling factors for library sizes and are automatically stored in `dge$samples$norm.factors` when working with DGEList objects.

Normalization is required to ensure that the expression distributions of each sample are similar across the entire experiment. The density plot for the filtered data above (right plot) shows that the distributions of log-CPM values are similar throughout all samples within the dataset. So, the effect of TMM normalization is mild as the magnitude of the scaling factors are all relatively close to 1.

```
dge <- calcNormFactors(dge, method = "TMM")
dge$samples$norm.factors
```

```
## [1] 1.0554452 1.0212137 0.9904567 0.9484696 1.0309324 0.9779832 1.0269341
## [8] 0.9538599
```

```
lcpm <- cpm(dge, log=TRUE)
boxplot(lcpm, las=2, col=col, main="")
title(main="Normalized data",ylab="Log-cpm")
```



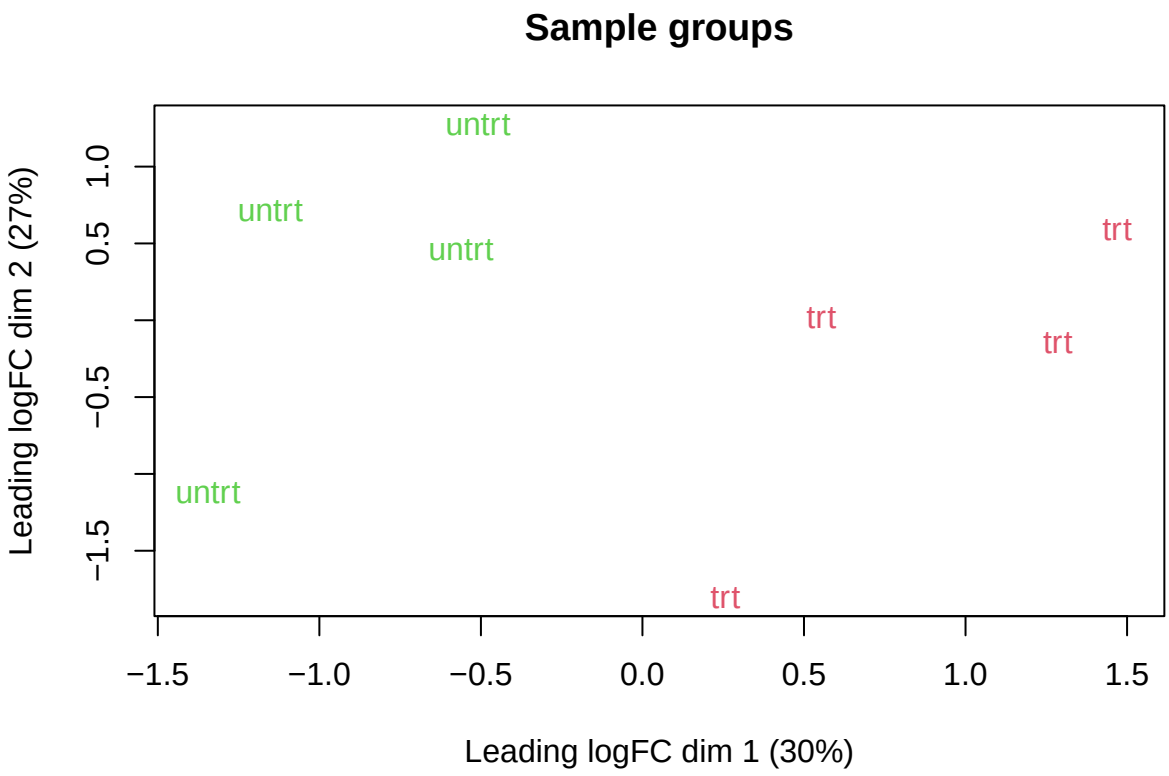
3.4 Unsupervised clustering of samples

Exploratory data analysis and visualization using multi-dimensional scaling (MDS) unsupervised technique and plotting using the **plotMDS** function in the **limma** package.

Used to glean insights into the extent to which differential expression can be detected before carrying out formal tests.

The plot shows similarities and dissimilarities between samples with well separated clusters between conditions. The first dimension represents the leading-fold-change that best separates samples and explains the largest proportion of variation in data, with subsequent orthogonal dimensions having a smaller effect.

```
limma::plotMDS(lcpm, labels=dge$samples$group, col=as.numeric(dge$samples$group)+1)
title(main="Sample groups")
```



Interactive MDS plots with Glimma package

```
glMDSPlot(lcpm, groups = dge$samples$group, launch = FALSE)
```

Link to interactive MDS plot

4. Differential Expression Analysis

To assess differential expression between the treated and untreated conditions, the **limma** package with the *voom* method, linear modeling and empirical Bayes moderation are used.

The **voom** (mean-variance modelling at the observational level) method is used to transform/process RNA-seq data prior to linear modeling in limma. It transforms raw count data (in the DGEList object) to log2-counts per million (logCPM), estimates the mean-variance relationship and uses this to compute appropriate observation-level weights. The data are then ready for linear modelling. The function estimates the mean-variance trend for log-counts, then assigns a weight to each observation based on its predicted variance. The weights are then used in the linear modelling process to adjust for non-constant variance in the residuals of the linear model.

With voom transformed RNA-seq raw counts to logCPM with associated precision weights, linear modeling in **limma** is carried out using **lmFit** function which produces a fitted model object containing coefficients, standard errors and residual standard errors for each gene.

lmFit has two main arguments, expression data and a design matrix which can be created using **model.matrix** base R function.

To make comparisons of interest once a linear model has been fit using an appropriate design matrix, the **makeContrasts** function is used to form a contrast matrix . The fit and the contrast matrix are then used

by **contrasts.fit** to compute fold changes and t-statistics for the comparisons/contrasts of interest: compute all possible pairwise comparisons between treated and untreated reference.

After fitting the linear model, the standard errors are moderated using an empirical Bayes model generated by the **eBayes** function to obtain more precise estimates of gene-wise variability. A moderated t-statistic and a log-odds of differential expression is computed for each contrast for each gene.

3.1 Creating a design matrix and contrasts

```
group <- dge$samples$group
design <- model.matrix(~0+group) # no intercept model
#design
colnames(design) <- gsub("group", "", colnames(design))
design
```

```
##   trt untrt
## 1   0     1
## 2   1     0
## 3   0     1
## 4   1     0
## 5   0     1
## 6   1     0
## 7   0     1
## 8   1     0
## attr(,"assign")
## [1] 1 1
## attr(,"contrasts")
## attr(,"contrasts")$group
## [1] "contr.treatment"
```

```
contr.matrix <- makeContrasts(
   trt_vs_untrt = trt-untrt,
   levels = colnames(design))
contr.matrix
```

```
##        Contrasts
## Levels  trt_vs_untrt
##   trt              1
##   untrt           -1
```

3.2 Assessing Differential Expression

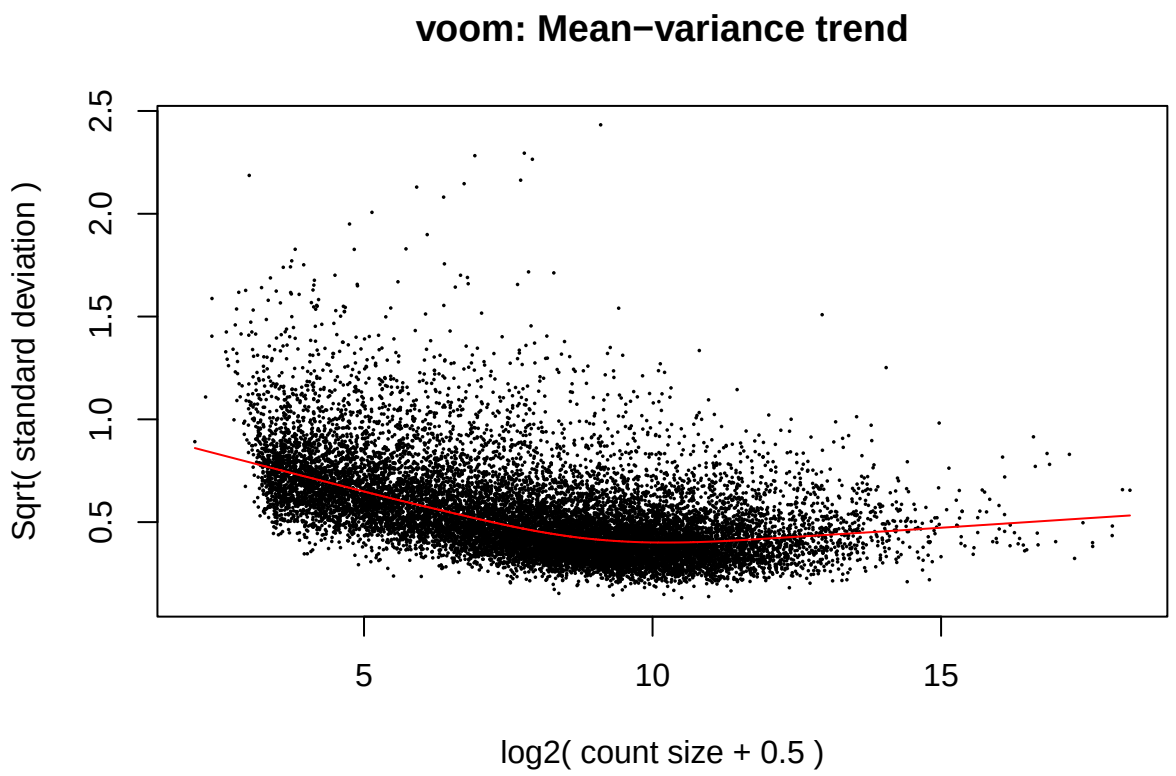
voom first converts the raw counts to logCPM values, adding 0.5 to all the counts to avoid taking the logarithm of zero. If counts is a DGEList object from the edgeR package, then voom will use the normalization factors found in the object when computing the logCPM values. In other words, the logCPM values are computed from the effective library sizes rather than the raw library sizes. The lmFit function is used to fit row-wise linear models. The lowess function is then used to fit a trend to the square-root-standard-deviations as a function of an average log-count measure. The trend line is then used to predict the variance of each logCPM value as a function of its fitted value on the count scale, and the inverse variances become the estimated precision weights.

Count data always show marked mean-variance relationships. Raw counts show increasing variance with increasing count size, while log-counts typically show a decreasing mean-variance trend.

The plot created within the voom function shows a decreasing trend between the means - the average log2 count for each gene, and quarter-root variances (square-root of standard deviations) which are the rescaled residual variances extracted by the voom function from fitting linear models to log-CPM transformed data. On the plot, each black dot represents a gene and the red curve shows the estimated mean-variance trend used to compute the voom weights.

voom generated an EList-object v retains the data frames stored within the DGEList-object that contain gene- and sample-level information: The `v$genes` data frame is equivalent to `dge$genes`, `v$targets` is equivalent to `dge$samples`, and the expression values stored in `v$E` is analogous to `dge$counts`, albeit on a transformed scale. In addition, the voom EList-object has the matrix of computed precision weights `v$weights` and stores the design matrix in `v$design`.

```
v <- voom(dge, design, plot=TRUE)
```




```
v
```

```
## An object of class "EList"
## $genes
##           ENSEMBL ENTREZID   SYMBOL TXCHROM
## 1 ENSG00000000003     7105   TSPAN6    chrX
## 3 ENSG00000000419     8813     DPM1   chr20
## 4 ENSG00000000457    57147    SCYL3    chr1
## 5 ENSG00000000460    55732 C1orf112    chr1
```

```
## 7 ENSG00000000971     3075      CFH    chr1
## 15921 more rows ...
##
## $targets
##            group lib.size norm.factors SampleName    cell   dex albut
## SRR1039508 untrt 21751039    1.0554452 GSM1275862  N61311 untrt untrt
## SRR1039509   trt 19181580    1.0212137 GSM1275863  N61311   trt untrt
## SRR1039512 untrt 25069766    0.9904567 GSM1275866 N052611 untrt untrt
## SRR1039513   trt 14364121    0.9484696 GSM1275867 N052611   trt untrt
## SRR1039516 untrt 25166984    1.0309324 GSM1275870 N080611 untrt untrt
## SRR1039517   trt 30098499    0.9779832 GSM1275871 N080611   trt untrt
## SRR1039520 untrt 19608386    1.0269341 GSM1275874 N061011 untrt untrt
## SRR1039521   trt 20160316    0.9538599 GSM1275875 N061011   trt untrt
##                     Run avgLength Experiment    Sample    BioSample
## SRR1039508 SRR1039508       126  SRX384345 SRS508568 SAMN02422669
## SRR1039509 SRR1039509       126  SRX384346 SRS508567 SAMN02422675
## SRR1039512 SRR1039512       126  SRX384349 SRS508571 SAMN02422678
## SRR1039513 SRR1039513        87  SRX384350 SRS508572 SAMN02422670
## SRR1039516 SRR1039516       120  SRX384353 SRS508575 SAMN02422682
## SRR1039517 SRR1039517       126  SRX384354 SRS508576 SAMN02422673
## SRR1039520 SRR1039520       101  SRX384357 SRS508579 SAMN02422683
## SRR1039521 SRR1039521        98  SRX384358 SRS508580 SAMN02422677
##
## $E
##                 SRR1039508 SRR1039509 SRR1039512 SRR1039513 SRR1039516
## ENSG00000000003   4.965317   4.547314  5.1227872   4.829794   5.499458
## ENSG00000000419   4.425810   4.748179  4.6317339   4.669330   4.544985
## ENSG00000000457   3.582127   3.462864  3.3937825   3.517546   3.286119
## ENSG00000000460   1.475851   1.532766  0.6919733   1.305349   1.641160
## ENSG00000000971   7.223877   7.583644  7.9449307   8.209698   8.061107
##                 SRR1039517 SRR1039520 SRR1039521
## ENSG00000000003   5.121115   5.296252   4.827685
## ENSG00000000419   4.731335   4.412233   4.656658
## ENSG00000000457   3.461245   3.573880   3.508904
## ENSG00000000460   1.077065   1.963989   1.585417
## ENSG00000000971   8.517198   8.044362   8.631526
## 15921 more rows ...
##
## $weights
##           [,1]      [,2]      [,3]      [,4]      [,5]     [,6]      [,7]
## [1,] 37.687122 34.064372 38.386535 30.296949 38.398866 37.98169 36.923732
## [2,] 32.835767 32.981438 34.526351 28.973119 34.570900 37.42485 31.464068
## [3,] 22.238216 20.714044 24.321607 16.898902 24.378976 27.41199 20.743677
## [4,]  8.383815  7.490192  9.213172  6.212996  9.237134 10.09871  7.833464
## [5,] 30.740082 29.585974 29.734075 31.656772 29.707087 26.66731 31.492829
##           [,8]
## [1,] 34.633034
## [2,] 33.611133
## [3,] 21.417309
## [4,]  7.735085
## [5,] 29.243561
## 15921 more rows ...
##
## $design
```

```
##   trt untrt
## 1   0     1
## 2   1     0
## 3   0     1
## 4   1     0
## 5   0     1
## 6   1     0
## 7   0     1
## 8   1     0
## attr(,"assign")
## [1] 1 1
## attr(,"contrasts")
## attr(,"contrasts")$group
## [1] "contr.treatment"
```

Final model

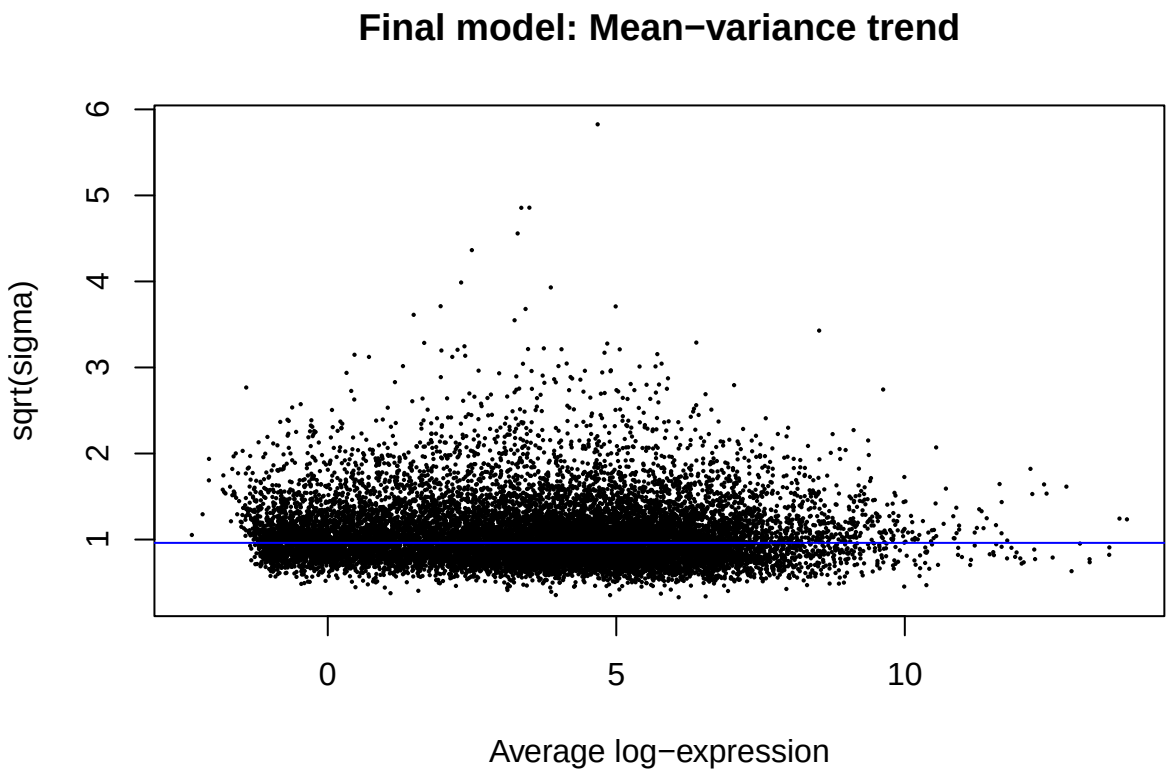
Given the linear model fit from **lmFit**, **eBayes** computes moderated t-statistics, moderated F-statistic, and log-odds of differential expression by empirical Bayes moderation of the standard errors squeezing the gene-wise residual variances towards a global value to rank genes in order of evidence for differential expression.

The empirical Bayes moderated t-statistics test each individual contrast equal to zero - test for genes that have true log-fold-changes different from zero. For each gene, the moderated F-statistic tests whether all the contrasts are zero. The F-statistic is an overall test computed from the set of t-statistics for that gene/probe.

plotSA function plots the quarter-root variance - the square-root of sigma, the estimated residual standard deviation from **contrasts.fit** function against mean log-CPM value. On the plot, the average log2 residual standard deviation estimated by the empirical Bayes algorithm is marked by a horizontal blue line.

The variance is no longer dependent on the mean expression level.

```
vfit <- lmFit(v, design)
vfit <- contrasts.fit(vfit, contrasts=contr.matrix)
efit <- eBayes(vfit)
plotSA(efit, main="Final model: Mean-variance trend")
```



Number of differentially expressed genes

```
vfit
```

```
## An object of class "MArrayLM"
## $coefficients
##                   Contrasts
##                    trt_vs_untrt
##   ENSG00000000003  -0.38163874
##   ENSG00000000419   0.19633949
##   ENSG00000000457   0.03286147
##   ENSG00000000460  -0.06692184
##   ENSG00000000971   0.41002291
## 15921 more rows ...
##
## $stdev.unscaled
##                   Contrasts
##                    trt_vs_untrt
##   ENSG00000000003    0.1179226
##   ENSG00000000419    0.1225388
##   ENSG00000000457    0.1499187
##   ENSG00000000460    0.2460776
##   ENSG00000000971    0.1294391
## 15921 more rows ...
```

```
##
## $sigma
## [1] 1.4111774 0.4634049 0.4989446 1.2330127 2.3799704
## 15921 more elements ...
##
## $df.residual
## [1] 6 6 6 6 6
## 15921 more elements ...
##
## $cov.coefficients
##               Contrasts
## Contrasts      trt_vs_untrt
##   trt_vs_untrt          0.5
##
## $pivot
## [1] 1 2
##
## $rank
## [1] 2
##
## $genes
##           ENSEMBL ENTREZID   SYMBOL TXCHROM
## 1 ENSG00000000003     7105   TSPAN6    chrX
## 3 ENSG00000000419     8813     DPM1   chr20
## 4 ENSG00000000457    57147    SCYL3    chr1
## 5 ENSG00000000460    55732 C1orf112    chr1
## 7 ENSG00000000971     3075      CFH    chr1
## 15921 more rows ...
##
## $Amean
## ENSG00000000003 ENSG00000000419 ENSG00000000457 ENSG00000000460 ENSG00000000971
##        5.026215        4.602533        3.473308        1.409196        8.027043
## 15921 more elements ...
##
## $method
## [1] "ls"
##
## $design
##   trt untrt
## 1   0     1
## 2   1     0
## 3   0     1
## 4   1     0
## 5   0     1
## 6   1     0
## 7   0     1
## 8   1     0
## attr(,"assign")
## [1] 1 1
## attr(,"contrasts")
## attr(,"contrasts")$group
## [1] "contr.treatment"
##
##
```

```
## $contrasts
##        Contrasts
## Levels  trt_vs_untrt
##   trt              1
##   untrt           -1
```

eBayes produces an object of class MArrayLM containing everything found in **contrasts.fit** `fit` MArrayLM object plus the following added components:

Component	Description
t	numeric matrix of moderated t-statistics.
p.value	numeric matrix of two-sided p-values corresponding to the t-statistics.
lods	numeric matrix giving the log-odds of differential expression (on the natural log scale).
s2.prior	estimated prior value for sigma^2.
df.prior	degrees of freedom associated with s2.prior.
df.total	row-wise numeric vector giving the total degrees of freedom associated with the t-statistics for each gene.
s2.post	row-wise numeric vector giving the posterior values for sigma^2.
var.prior	column-wise numeric vector giving estimated prior values for the variance of the log2-fold-changes for differentially expressed gene for each contrast. Used for evaluating lods.
F	row-wise numeric vector of moderated F-statistics for testing all contrasts defined by the columns of fit simultaneously equal to zero.
F.p.value	row-wise numeric vector giving p-values corresponding to F.

The matrices t, p.value and lods have the same dimensions as the input object `fit`, with rows corresponding to genes and columns to contrasts. The vectors s2.prior, df.prior, df.total, F and F.p.value correspond to rows, with length equal to the number of genes. The vector var.prior corresponds to columns, with length equal to the number of contrasts. If s2.prior or df.prior have length 1, then the same value applies to all genes.

s2.prior, df.prior and var.prior contain empirical Bayes hyperparameters used to obtain df.total, s2.post and lods.

Notes: Empirical Bayes (also known as maximum marginal likelihood) method is a procedure for statistical inference in which the prior probability distribution is estimated from the data before creating a posterior probability distribution. A convenient approach for setting model hyperparameters, that is the priors distribution, instead of fixed values for each parameter in a prior assumption.

```
efit
```

```
## An object of class "MArrayLM"
## $coefficients
##                   Contrasts
##                    trt_vs_untrt
##   ENSG00000000003  -0.38163874
##   ENSG00000000419   0.19633949
##   ENSG00000000457   0.03286147
##   ENSG00000000460  -0.06692184
##   ENSG00000000971   0.41002291
## 15921 more rows ...
##
```

```
## $stdev.unscaled
##                   Contrasts
##                    trt_vs_untrt
##   ENSG00000000003    0.1179226
##   ENSG00000000419    0.1225388
##   ENSG00000000457    0.1499187
##   ENSG00000000460    0.2460776
##   ENSG00000000971    0.1294391
## 15921 more rows ...
##
## $sigma
## [1] 1.4111774 0.4634049 0.4989446 1.2330127 2.3799704
## 15921 more elements ...
##
## $df.residual
## [1] 6 6 6 6 6
## 15921 more elements ...
##
## $cov.coefficients
##               Contrasts
## Contrasts      trt_vs_untrt
##   trt_vs_untrt          0.5
##
## $pivot
## [1] 1 2
##
## $rank
## [1] 2
##
## $genes
##           ENSEMBL ENTREZID   SYMBOL TXCHROM
## 1 ENSG00000000003     7105   TSPAN6    chrX
## 3 ENSG00000000419     8813     DPM1   chr20
## 4 ENSG00000000457    57147    SCYL3    chr1
## 5 ENSG00000000460    55732 C1orf112    chr1
## 7 ENSG00000000971     3075      CFH    chr1
## 15921 more rows ...
##
## $Amean
## ENSG00000000003 ENSG00000000419 ENSG00000000457 ENSG00000000460 ENSG00000000971
##        5.026215        4.602533        3.473308        1.409196        8.027043
## 15921 more elements ...
##
## $method
## [1] "ls"
##
## $design
##   trt untrt
## 1   0     1
## 2   1     0
## 3   0     1
## 4   1     0
## 5   0     1
## 6   1     0
```

```
## 7   0     1
## 8   1     0
## attr(,"assign")
## [1] 1 1
## attr(,"contrasts")
## attr(,"contrasts")$group
## [1] "contr.treatment"
##
##
## $contrasts
##        Contrasts
## Levels  trt_vs_untrt
##   trt              1
##   untrt           -1
##
## $df.prior
## [1] 2.847972
##
## $s2.prior
## [1] 0.8536935
##
## $var.prior
## [1] 2.382541
##
## $proportion
## [1] 0.01
##
## $s2.post
## [1] 1.6252114 0.4204082 0.4436010 1.3057475 4.1158412
## 15921 more elements ...
##
## $t
##                  Contrasts
##                   trt_vs_untrt
##   ENSG00000000003   -2.5386372
##   ENSG00000000419    2.4711471
##   ENSG00000000457    0.3291053
##   ENSG00000000460   -0.2379941
##   ENSG00000000971    1.5613970
## 15921 more rows ...
##
## $df.total
## [1] 8.847972 8.847972 8.847972 8.847972 8.847972
## 15921 more elements ...
##
## $p.value
##                  Contrasts
##                   trt_vs_untrt
##   ENSG00000000003   0.03219035
##   ENSG00000000419   0.03593099
##   ENSG00000000457   0.74973228
##   ENSG00000000460   0.81730549
##   ENSG00000000971   0.15344160
## 15921 more rows ...
```

```
##
## $lods
##                   Contrasts
##                    trt_vs_untrt
##   ENSG00000000003    -4.496281
##   ENSG00000000419    -4.568672
##   ENSG00000000457    -6.872214
##   ENSG00000000460    -6.413221
##   ENSG00000000971    -5.888373
## 15921 more rows ...
##
## $F
## [1] 6.44467869 6.10656816 0.10831029 0.05664118 2.43796073
## 15921 more elements ...
##
## $F.p.value
## [1] 0.03219035 0.03593099 0.74973228 0.81730549 0.15344160
## 15921 more elements ...
```

The number of significantly up- and down-regulated genes can be summaries in a table using the results of the **decideTests** function which performs multiple testing across genes and contrasts and identifies which genes are significantly differentially expressed for each contrast from the eBayes `fit` object containing p-values and test statistics. The function returns an object of class TestResults which is essentially a numeric matrix with elements -1, 0 or 1 depending on whether each t-statistic is classified as significantly negative, not significant or significantly positive. Significance is defined using an adjusted p-value cutoff that is set at 5% by default. The adjust method used is false discovery rate Benjamini-Hochberg, "BH" or "fdr".

```
dt <- decideTests(efit, method = "separate",
                  adjust.method = "BH", p.value = 0.05, lfc = 0)
summary(dt)
```

```
##        trt_vs_untrt
## Down            845
## NotSig        14048
## Up             1033
```

Table of top genes from eBayes model fit

```
tt <- topTable(efit, coef = 1, number = Inf, genelist = efit$genes,
         adjust.method = "BH", p.value = 0.05, lfc=0,
         sort.by = "B") # B-log-odds that the gene is differentially expressed
dim(tt)
```

```
## [1] 1878   10
```

```
head(tt, n=20)
```

```
##                         ENSEMBL ENTREZID  SYMBOL TXCHROM     logFC  AveExpr
## ENSG00000152583 ENSG00000152583     8404 SPARCL1    chr4  4.574574 4.167526
## ENSG00000134686 ENSG00000134686     1912    PHC2    chr1  1.380829 6.839533
## ENSG00000125148 ENSG00000125148     4502    MT2A   chr16  2.198850 7.024695
## ENSG00000148175 ENSG00000148175     2040    STOM    chr9  1.440250 8.856653
```

```
## ENSG00000179094 ENSG00000179094     5187    PER1   chr17  3.185448 4.421045
## ENSG00000120129 ENSG00000120129     1843   DUSP1    chr5  2.950668 6.645078
## ENSG00000178695 ENSG00000178695   115207  KCTD12   chr13 -2.530760 6.452348
## ENSG00000183044 ENSG00000183044       18    ABAT   chr16  1.148516 4.432683
## ENSG00000189221 ENSG00000189221     4128    MAOA    chrX  3.333785 5.951182
## ENSG00000162616 ENSG00000162616    11080  DNAJB4    chr1  1.508948 5.905545
## ENSG00000162614 ENSG00000162614    91624    NEXN    chr1  2.034649 7.638473
## ENSG00000196517 ENSG00000196517     6536  SLC6A9    chr1 -2.215413 3.026954
## ENSG00000101347 ENSG00000101347    25939  SAMHD1   chr20  3.780495 8.135879
## ENSG00000144369 ENSG00000144369   165215 FAM171B    chr2 -1.357174 5.775168
## ENSG00000096060 ENSG00000096060     2289   FKBP5    chr6  4.009429 5.781384
## ENSG00000116584 ENSG00000116584     9181 ARHGEF2    chr1 -1.026536 6.630742
## ENSG00000181467 ENSG00000181467     5912   RAP2B    chr3 -1.272100 4.746027
## ENSG00000077684 ENSG00000077684    79960   JADE1    chr4  1.196813 6.074470
## ENSG00000164647 ENSG00000164647    26872  STEAP1    chr7  1.562268 6.282059
## ENSG00000114098 ENSG00000114098    25852   ARMC8    chr3  1.340071 5.405407
##                         t      P.Value    adj.P.Val        B
## ENSG00000152583  18.57076 2.159409e-08 0.0001912819 9.421860
## ENSG00000134686  16.92813 4.801163e-08 0.0001912819 9.251227
## ENSG00000125148  16.49336 6.005335e-08 0.0001912819 9.030024
## ENSG00000148175  16.63751 5.572491e-08 0.0001912819 9.000249
## ENSG00000179094  16.59375 5.700111e-08 0.0001912819 8.927846
## ENSG00000120129  15.80840 8.643069e-08 0.0002261997 8.688279
## ENSG00000178695 -15.01752 1.341060e-07 0.0002261997 8.274343
## ENSG00000183044  14.96605 1.380978e-07 0.0002261997 8.229756
## ENSG00000189221  14.83766 1.486416e-07 0.0002261997 8.167530
## ENSG00000162616  14.56317 1.743137e-07 0.0002261997 8.035881
## ENSG00000162614  14.46516 1.846412e-07 0.0002261997 7.959000
## ENSG00000196517 -15.08382 1.291515e-07 0.0002261997 7.949962
## ENSG00000101347  14.48813 1.821609e-07 0.0002261997 7.948880
## ENSG00000144369 -14.04978 2.366164e-07 0.0002615271 7.736820
## ENSG00000096060  13.75759 2.828722e-07 0.0002815639 7.537519
## ENSG00000116584 -13.64774 3.027858e-07 0.0002836568 7.494384
## ENSG00000181467 -13.08290 4.330996e-07 0.0003644267 7.139044
## ENSG00000077684  13.07695 4.347676e-07 0.0003644267 7.136982
## ENSG00000164647  12.69830 5.571536e-07 0.0004127952 6.890529
## ENSG00000114098  12.66340 5.702307e-07 0.0004127952 6.867532
```

Use the **treat** method to calculate p-values from a eBayes moderated t-statistics with a minimum logFC requirement. Instead of testing for genes that have true log-fold-changes different from zero, it tests whether the true log2-fold-change is greater than lfc in absolute value. When the number of DE genes is large, treat is often useful for giving preference to larger fold-changes and for prioritizing genes that are biologically important. treat is concerned with p-values rather than posterior odds.

```
tfit <- treat(vfit, lfc=1)
dt <- decideTests(tfit)
summary(dt)
```

```
##        trt_vs_untrt
## Down              7
## NotSig        15896
## Up               23
```

Table of top genes

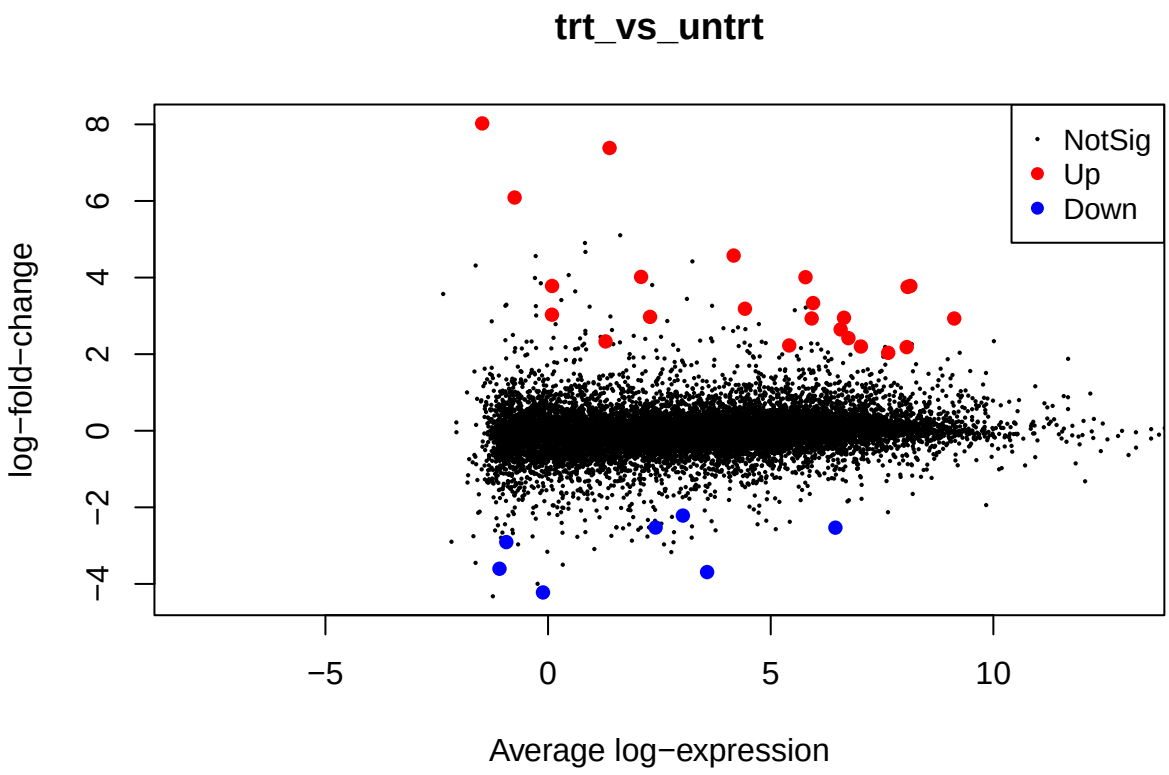
```
trt_vs_untrt <- topTreat(tfit, coef=1, n=Inf)
head(trt_vs_untrt, n=30)
```

```
##                         ENSEMBL ENTREZID   SYMBOL TXCHROM     logFC     AveExpr
## ENSG00000152583 ENSG00000152583     8404  SPARCL1    chr4  4.574574  4.16752647
## ENSG00000179593 ENSG00000179593      247  ALOX15B   chr17  8.025059 -1.47925282
## ENSG00000179094 ENSG00000179094     5187     PER1   chr17  3.185448  4.42104464
## ENSG00000101347 ENSG00000101347    25939   SAMHD1   chr20  3.780495  8.13587918
## ENSG00000120129 ENSG00000120129     1843    DUSP1    chr5  2.950668  6.64507758
## ENSG00000189221 ENSG00000189221     4128     MAOA    chrX  3.333785  5.95118194
## ENSG00000096060 ENSG00000096060     2289    FKBP5    chr6  4.009429  5.78138384
## ENSG00000178695 ENSG00000178695   115207   KCTD12   chr13 -2.530760  6.45234766
## ENSG00000125148 ENSG00000125148     4502     MT2A   chr16  2.198850  7.02469486
## ENSG00000196517 ENSG00000196517     6536   SLC6A9    chr1 -2.215413  3.02695429
## ENSG00000250978 ENSG00000250978     <NA>     <NA>    <NA>  6.090261 -0.75155961
## ENSG00000198624 ENSG00000198624    26112   CCDC69    chr5  2.931597  5.91928585
## ENSG00000162692 ENSG00000162692     7412    VCAM1    chr1 -3.690477  3.57158279
## ENSG00000135821 ENSG00000135821     2752     GLUL    chr1  2.930880  9.12073543
## ENSG00000162614 ENSG00000162614    91624     NEXN    chr1  2.034649  7.63847305
## ENSG00000158246 ENSG00000158246   115572   TENT5B    chr1  2.332101  1.29098843
## ENSG00000143494 ENSG00000143494    79805    VASH2    chr1 -3.603602 -1.09164048
## ENSG00000146006 ENSG00000146006    26045   LRRTM2    chr5 -4.222527 -0.11486369
## ENSG00000170214 ENSG00000170214      147   ADRA1B    chr5  4.017179  2.08880692
## ENSG00000163491 ENSG00000163491   152110    NEK10    chr3 -2.527799  2.41439987
## ENSG00000166741 ENSG00000166741     4837     NNMT   chr11  2.183864  8.05401154
## ENSG00000211445 ENSG00000211445     2878     GPX3    chr5  3.755618  8.07422840
## ENSG00000109906 ENSG00000109906     7704   ZBTB16   chr11  7.383926  1.38134435
## ENSG00000139132 ENSG00000139132   121512     FGD4   chr12  2.226778  5.41596363
## ENSG00000157150 ENSG00000157150     7079    TIMP4    chr3  2.973499  2.29028307
## ENSG00000103196 ENSG00000103196    83716 CRISPLD2   chr16  2.645297  6.57072881
## ENSG00000116285 ENSG00000116285    54206   ERRFI1    chr1  2.417881  6.74491185
## ENSG00000122877 ENSG00000122877     1959     EGR2   chr10 -2.907873 -0.93829615
## ENSG00000173838 ENSG00000173838   162333 MARCHF10   chr17  3.780102  0.08978235
## ENSG00000187193 ENSG00000187193     4501     MT1X   chr16  3.030425  0.08674743
##                         t      P.Value   adj.P.Val
## ENSG00000152583 14.511198 9.179440e-08 0.001461918
## ENSG00000179593 12.241029 4.244896e-07 0.003380211
## ENSG00000179094 11.384514 6.986468e-07 0.003601008
## ENSG00000101347 10.655795 1.217356e-06 0.003601008
## ENSG00000120129 10.450834 1.418301e-06 0.003601008
## ENSG00000189221 10.386969 1.495920e-06 0.003601008
## ENSG00000096060 10.326281 1.582761e-06 0.003601008
## ENSG00000178695 -9.083523 4.446559e-06 0.008529022
## ENSG00000125148  8.992460 4.819867e-06 0.008529022
## ENSG00000196517 -8.275236 9.366068e-06 0.014916401
## ENSG00000250978  8.009094 1.294655e-05 0.018292019
## ENSG00000198624  7.882125 1.378276e-05 0.018292019
## ENSG00000162692 -7.804274 1.501181e-05 0.018390619
## ENSG00000135821  7.508143 2.014011e-05 0.022910812
## ENSG00000162614  7.355747 2.353795e-05 0.024991031
## ENSG00000158246  7.267537 2.584055e-05 0.025721034
## ENSG00000143494 -7.212179 2.765955e-05 0.025912114
## ENSG00000146006 -6.864030 4.079073e-05 0.036090731
```

```
## ENSG00000170214  6.810643 4.310499e-05 0.036131055
## ENSG00000163491 -6.746841 4.554709e-05 0.036269147
## ENSG00000166741  6.695800 4.818931e-05 0.036545855
## ENSG00000211445  6.603169 5.417495e-05 0.037391388
## ENSG00000109906  6.686663 5.450406e-05 0.037391388
## ENSG00000139132  6.557621 5.634769e-05 0.037391388
## ENSG00000157150  6.497072 6.058155e-05 0.038140925
## ENSG00000103196  6.471453 6.226699e-05 0.038140925
## ENSG00000116285  6.413007 6.655625e-05 0.039258325
## ENSG00000122877 -6.302586 7.586876e-05 0.043153066
## ENSG00000173838  6.260948 8.052147e-05 0.044220169
## ENSG00000187193  6.130809 9.304738e-05 0.049395750
```

Mean-difference plot summarizing results for all genes and displaying logFCs from the linear model fit against the average log-CPM values with differentially expressed genes highlighted.

```
plotMD(tfit, column=1, status=dt[,1], main=colnames(tfit)[1],
       xlim=c(-8,13))
```



Interactive mean-difference plot with Glimma glMDPlot

```
glMDPlot(tfit, coef=1, status=dt, main=colnames(tfit)[1],
         side.main="SYMBOL", counts=lcpm, groups=group, launch=FALSE)
```

The output HTML page includes summarized results (logFCs versus log-CPM values) in the left panel which is linked to individual log-CPM values per sample for a selected gene in the right panel. A table of results

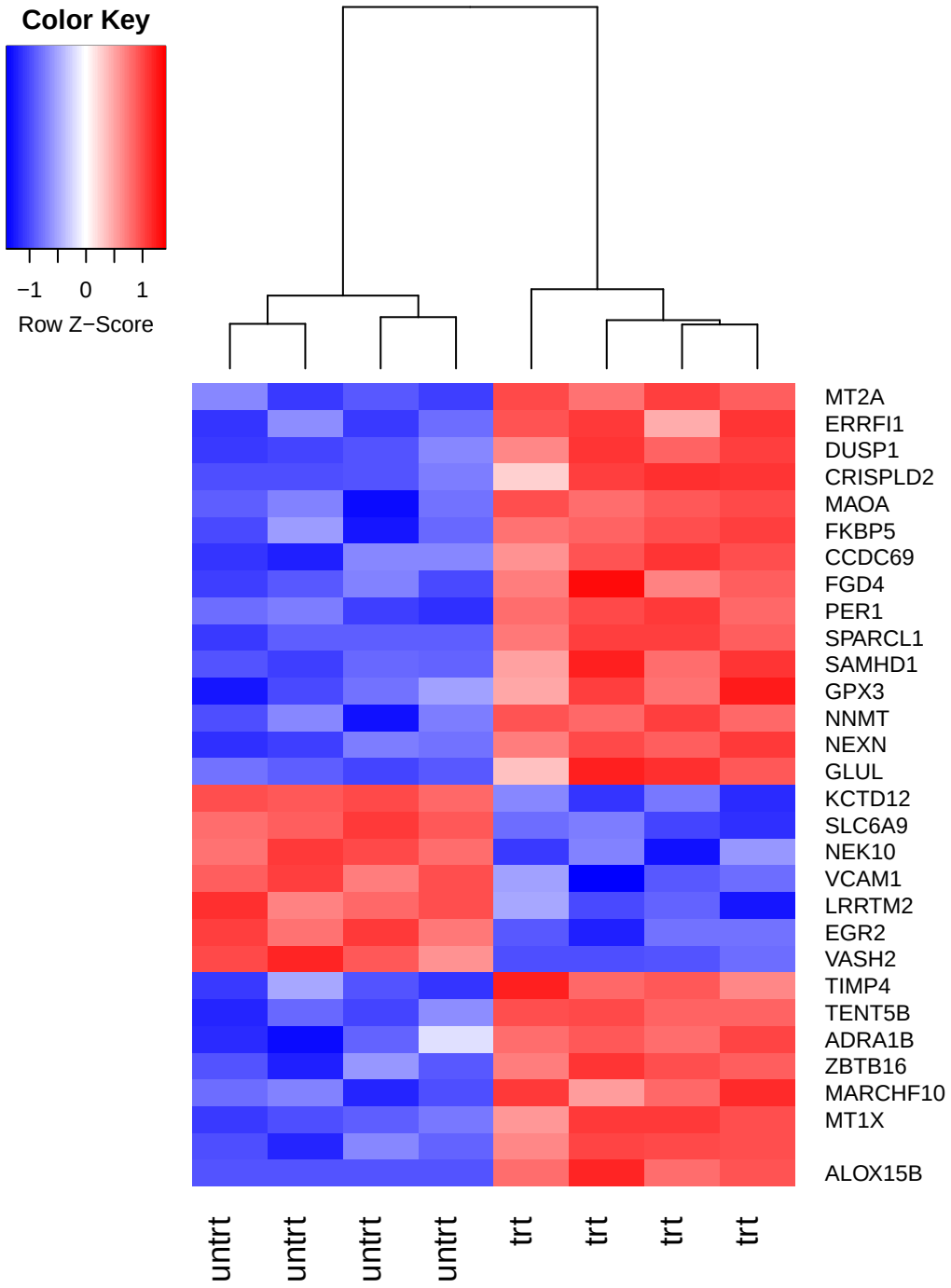
is also displayed below these figures, along with a search bar for looking up a particular gene based on the annotation information available (e.g. gene symbol identifier).

Link to interactive MD plot

Glimma is implemented in R and Javascript, with the R code generated the data which is converted into graphics the D3 Javascript library, with the Bootstrap library handling layouts and Datatables generating the interactive searchable tables.

Heatmap of log-CPM values for to 10 genes differentially expressed in treated versus untreated ranked by adjusted p-value

```
trt_vs_untrt.topgenes <- trt_vs_untrt$ENSEMBL[1:30]
i <- which(v$genes$ENSEMBL %in% trt_vs_untrt.topgenes)
mycol <- colorpanel(100,"blue","white","red")
heatmap.2(lcpm[i,], scale="row",
   labRow=v$genes$SYMBOL[i], labCol=group,
   col=mycol, trace="none", density.info="none",
   margin=c(8,6), dendrogram="column")
```



The results obtained from this analysis correlate well with findings from published literature PMID: 24926665. The top differentially expressed genes include DUSP1, PER1, CCDC69 and CRISPLD2 glucocorticoid-responsive genes with increased expression in dexamethasone treated cells.

```
sessionInfo()
```

```
## R version 4.2.1 (2022-06-23)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Debian GNU/Linux 11 (bullseye)
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.9.0
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.9.0
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats4    stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] RColorBrewer_1.1-3
##  [2] gplots_3.1.3
##  [3] airway_1.16.0
##  [4] SummarizedExperiment_1.26.1
##  [5] MatrixGenerics_1.8.1
##  [6] matrixStats_0.62.0
##  [7] Homo.sapiens_1.3.1
##  [8] TxDb.Hsapiens.UCSC.hg19.knownGene_3.2.2
##  [9] org.Hs.eg.db_3.15.0
## [10] GO.db_3.15.0
## [11] OrganismDbi_1.38.1
## [12] GenomicFeatures_1.48.4
## [13] GenomicRanges_1.48.0
## [14] GenomeInfoDb_1.32.4
## [15] AnnotationDbi_1.58.0
## [16] IRanges_2.30.1
## [17] S4Vectors_0.34.0
## [18] Biobase_2.56.0
## [19] BiocGenerics_0.42.0
## [20] edgeR_3.38.4
## [21] Glimma_2.6.0
## [22] limma_3.52.4
##
## loaded via a namespace (and not attached):
##  [1] colorspace_2.0-3        rjson_0.2.21            ellipsis_0.3.2
##  [4] XVector_0.36.0          rstudioapi_0.14         bit64_4.0.5
##  [7] fansi_1.0.3             xml2_1.3.3              codetools_0.2-18
## [10] splines_4.2.1           cachem_1.0.6            geneplotter_1.74.0
## [13] knitr_1.40              jsonlite_1.8.2          Rsamtools_2.12.0
## [16] annotate_1.74.0         dbplyr_2.2.1            png_0.1-7
```

```
## [19] graph_1.74.0             BiocManager_1.30.18      compiler_4.2.1
## [22] httr_1.4.4               assertthat_0.2.1         Matrix_1.3-2
## [25] fastmap_1.1.0            cli_3.4.1                htmltools_0.5.3
## [28] prettyunits_1.1.1        tools_4.2.1              gtable_0.3.1
## [31] glue_1.6.2               GenomeInfoDbData_1.2.8   dplyr_1.0.10
## [34] rappdirs_0.3.3           Rcpp_1.0.9               vctrs_0.4.2
## [37] Biostrings_2.64.1        rtracklayer_1.56.1       xfun_0.33
## [40] stringr_1.4.1            lifecycle_1.0.3          restfulr_0.0.15
## [43] gtools_3.9.3             XML_3.99-0.11            zlibbioc_1.42.0
## [46] scales_1.2.1             hms_1.1.2                parallel_4.2.1
## [49] RBGL_1.72.0              yaml_2.3.5               curl_4.3.3
## [52] memoise_2.0.1            ggplot2_3.3.6            biomaRt_2.52.0
## [55] stringi_1.7.8            RSQLite_2.2.18           highr_0.9
## [58] genefilter_1.78.0        BiocIO_1.6.0             caTools_1.18.2
## [61] filelock_1.0.2           BiocParallel_1.30.4      rlang_1.0.6
## [64] pkgconfig_2.0.3          bitops_1.0-7             evaluate_0.17
## [67] lattice_0.20-41          GenomicAlignments_1.32.1 htmlwidgets_1.5.4
## [70] bit_4.0.4                tidyselect_1.2.0         magrittr_2.0.3
## [73] DESeq2_1.36.0            R6_2.5.1                 generics_0.1.3
## [76] DelayedArray_0.22.0      DBI_1.1.3                pillar_1.8.1
## [79] survival_3.2-7           KEGGREST_1.36.3          RCurl_1.98-1.9
## [82] tibble_3.1.8             crayon_1.5.2             KernSmooth_2.23-18
## [85] utf8_1.2.2               BiocFileCache_2.4.0      rmarkdown_2.17
## [88] progress_1.2.2           locfit_1.5-9.6           grid_4.2.1
## [91] blob_1.2.3               digest_0.6.29            xtable_1.8-4
## [94] munsell_0.5.0
```