maketitle

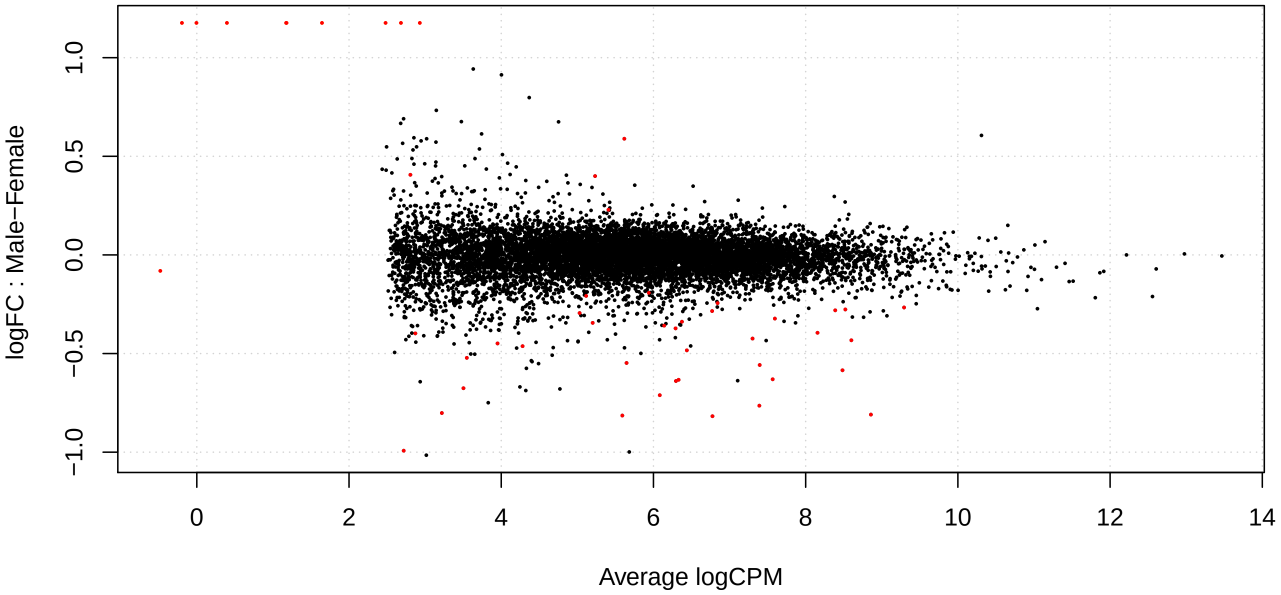
Contents

```
library(DEGreport)
data(humanSexDEedgeR)
library(edgeR)
```

1 QC figures from DE analysis

We are going to do a differential expression analysis with edgeR. We have an object that is comming from the edgeR package. It countains a gene count matrix for 85 TSI HapMap individuals, and the gender information. With that, we are going to apply the ‘glmFit‘ function to get genes differentially expressed between males and females.

```
des<-humanSexDEedgeR$design
fit <- glmFit(humanSexDEedgeR,des)
lrt <- glmLRT(fit)
tab<-cbind(lrt$table,p.adjust(lrt$table$PValue,method="BH"))
detags <- rownames(tab[tab[,5]<=0.1,])
plotSmear(humanSexDEedgeR, de.tags=detags)
```



We need to extract the experiment design data.frame where the condition is Male or Female.

```
counts<-cpm(humanSexDEedgeR,log=FALSE)
g1<-colnames(counts)[1:41]
g2<-colnames(counts)[42:85]
design<-data.frame(condition=sub("1","Male",sub("0","Female",des[,2])))
```

We are getting the chromosome information for each gene. This way we can colour genes according autosomic,X or Y chromosomes.

```
data(geneInfo)
```

Create the report. The main parameters are the column names in group1, and group2. Then, the count matrix, gene names that are DE, p-values, fold changes and path to create the report. As optional, you can give colours for each gene, and the number of permutation.

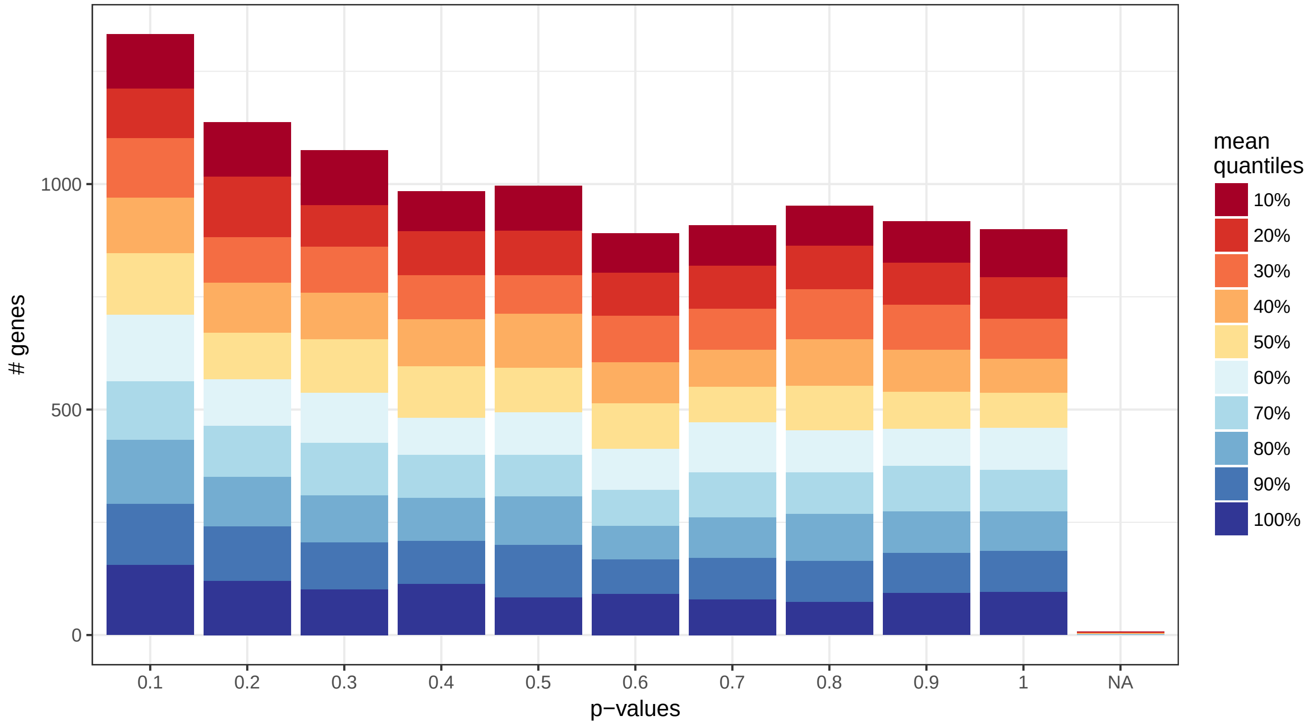
```
detag10<-detags[1:10]
pval<-tab[,4]
fc<-tab[detag10,1]
```

Run the fowlling lines if you want to visualize your expression values by condition:

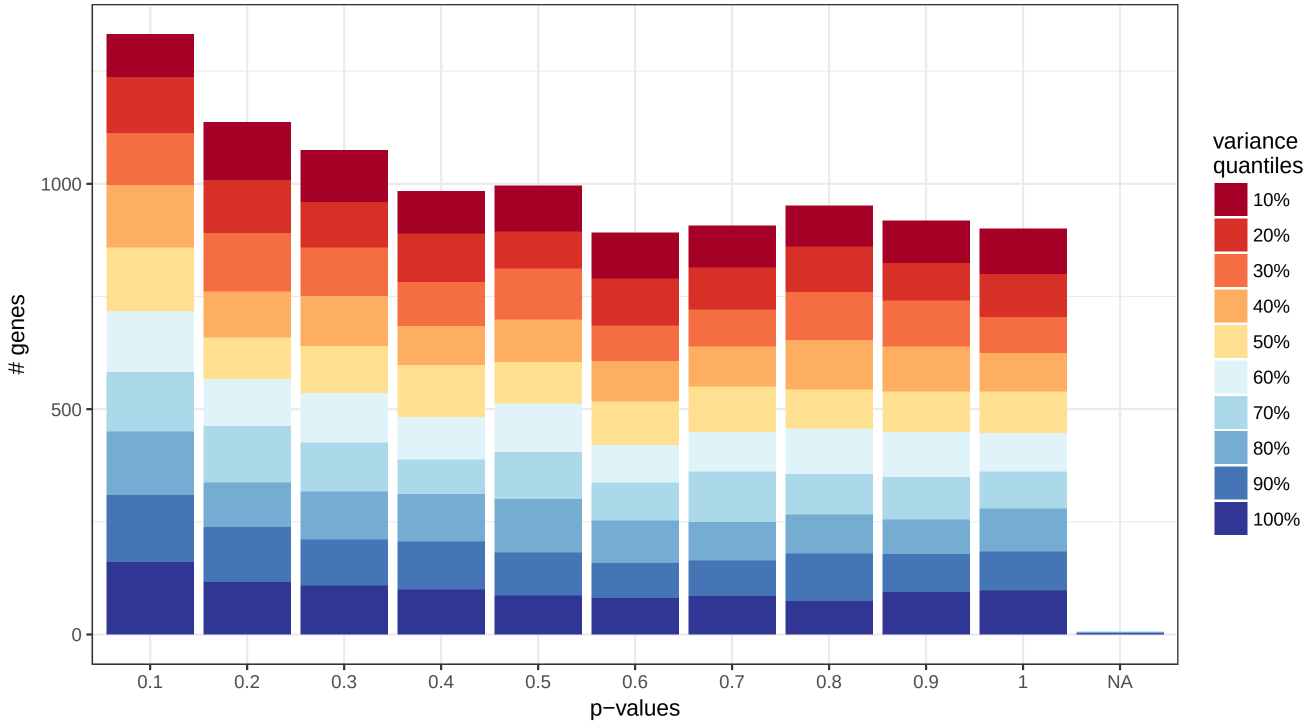
```
degObj(counts,design,"degObj.rda")
library(shiny)
runGist(9930881)
```

You can use individual functions, like degRank or degMean. This will create specific figures and tables that are included in the report.

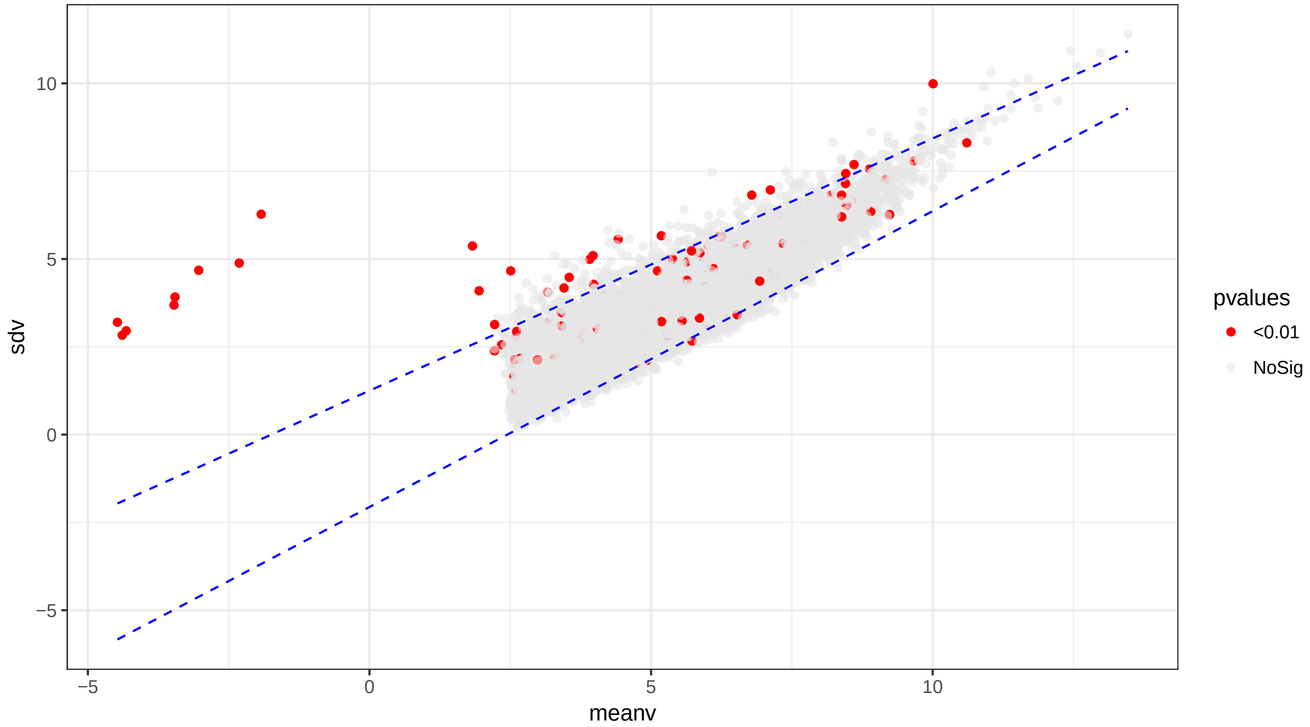
```
degMean(pval,counts)
```



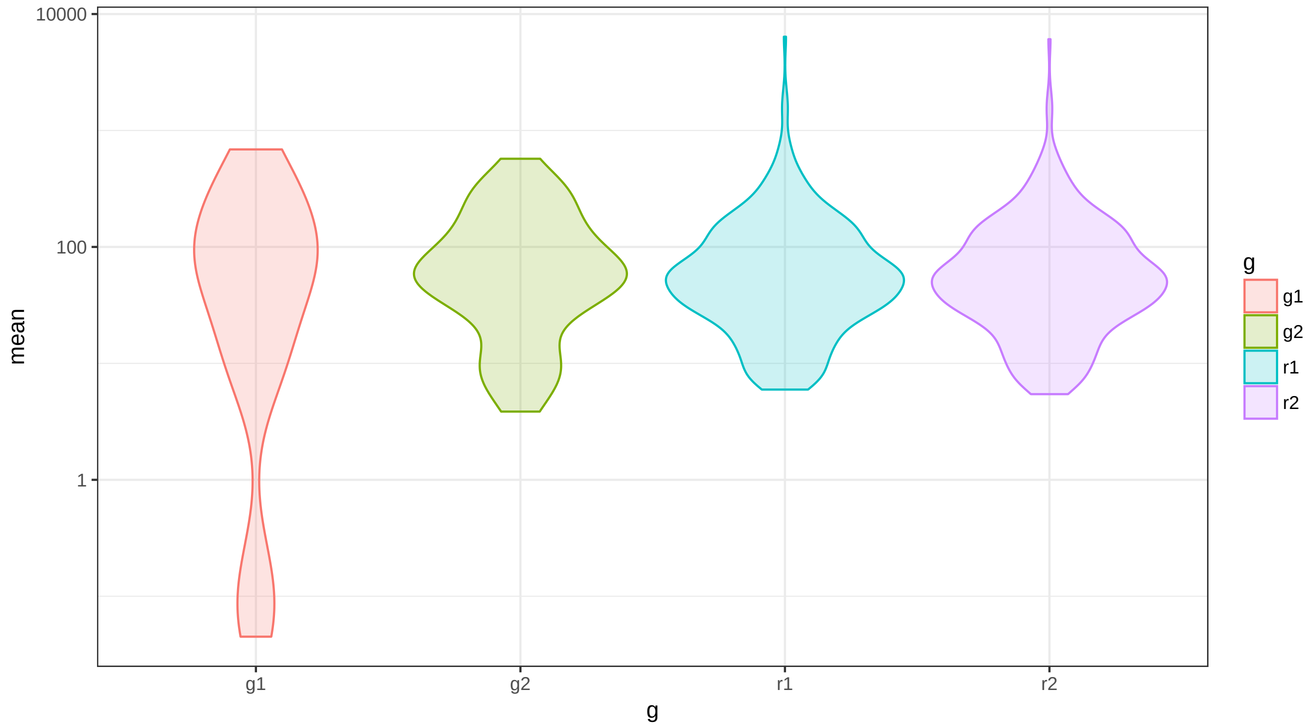
```
degVar(pval,counts)
```



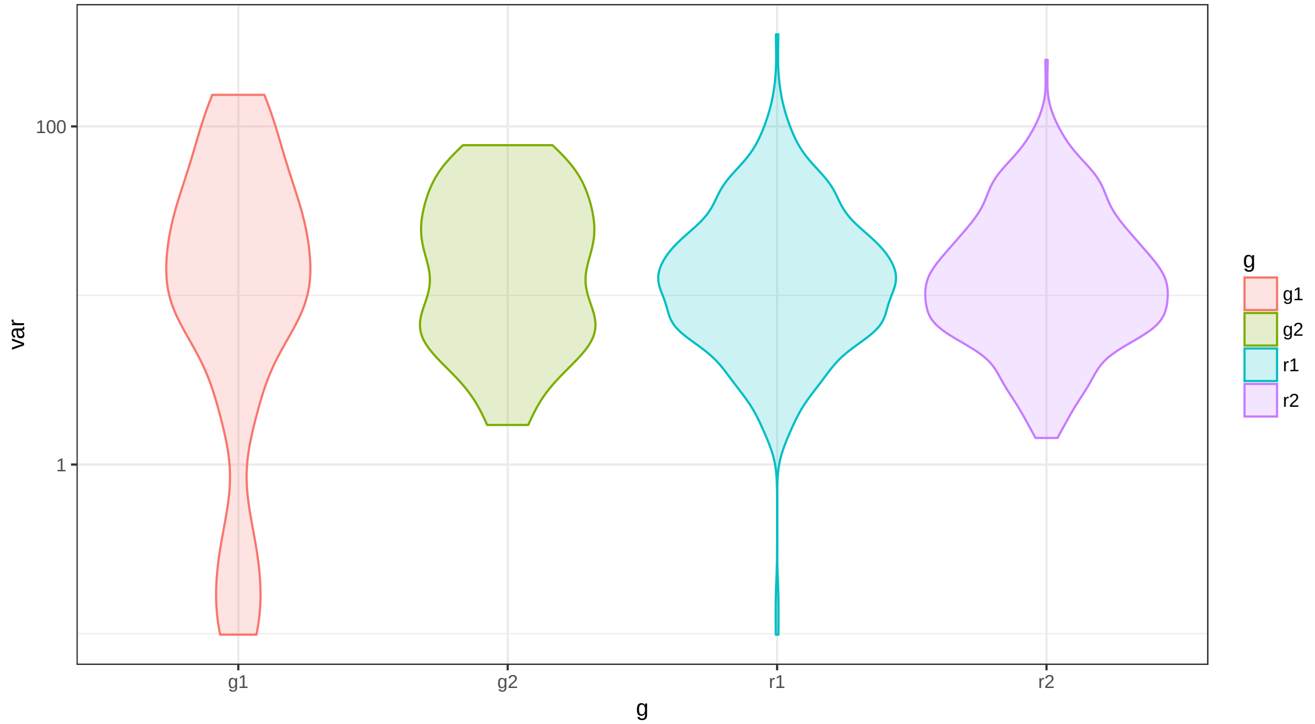
```
degMV(humanSexDEedgeR$samples$group,pval,counts)
```



```
degMB(detags,g1,g2,counts)
```



```
degVB(detags,g1,g2,counts)
```



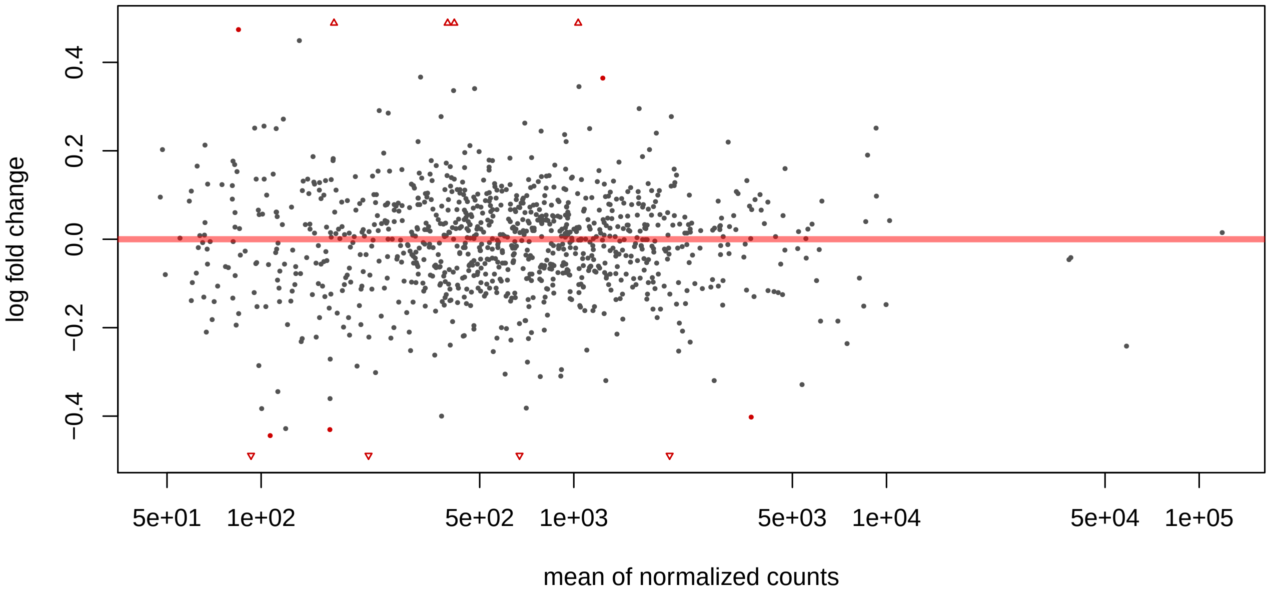
```
# require(rjags)
# rank<-degRank(g1,g2,counts[detag10,],fc,400,500)
# degPR(rank)
```

2 Report from DESeq2 analysis

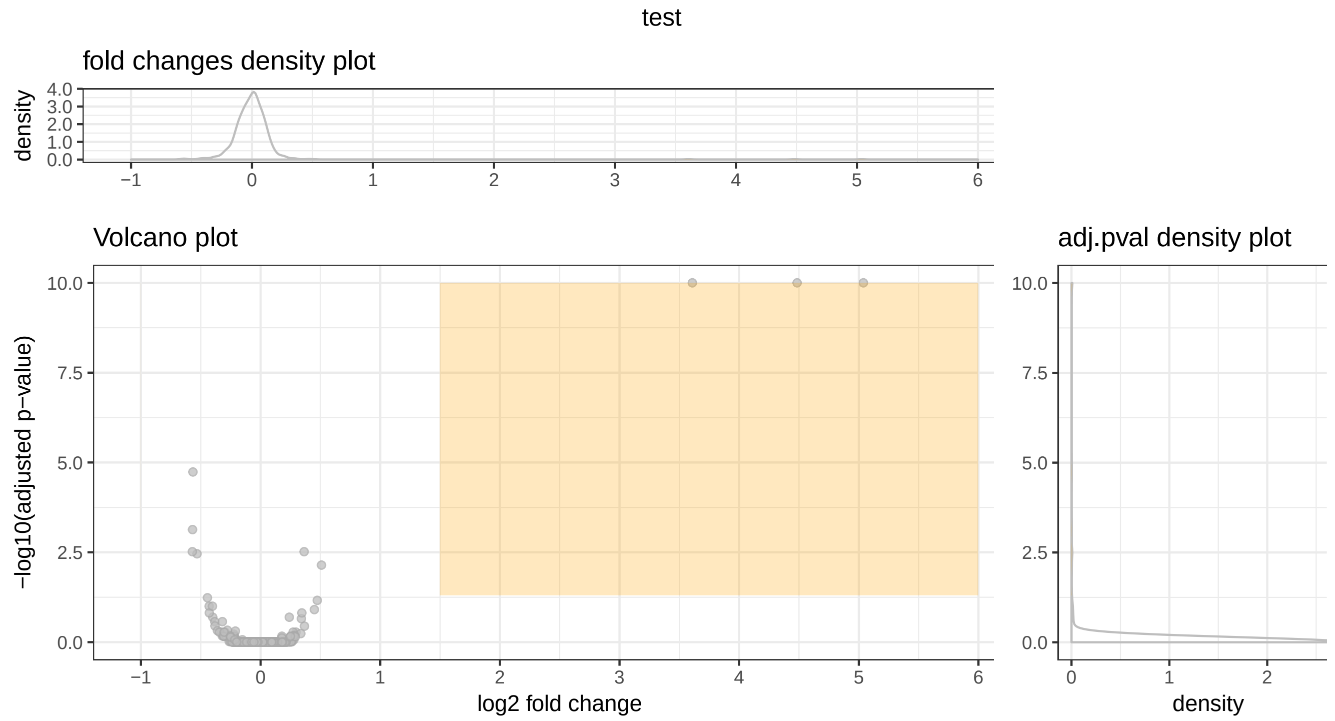
In this section, we show how to use DESeq2 output to create a full report, including figures and tbale with top deregulated genes, GO enrichment analysis and heatmaps and PCA plots. If you set `path_results`, different files will be saved there.

```
data(humanSexDEedgeR)
library(DESeq2)
idx <- c(1:10, 75:85)
dse <- DESeqDataSetFromMatrix(humanSexDEedgeR$counts[1:1000, idx],
humanSexDEedgeR$samples[idx,], design=~group)
dse <- DESeq(dse)
res <- degResults(dds=dse, name="test", org=NULL,
do_go=FALSE, group="group", xs="group", path_results = NULL)
```

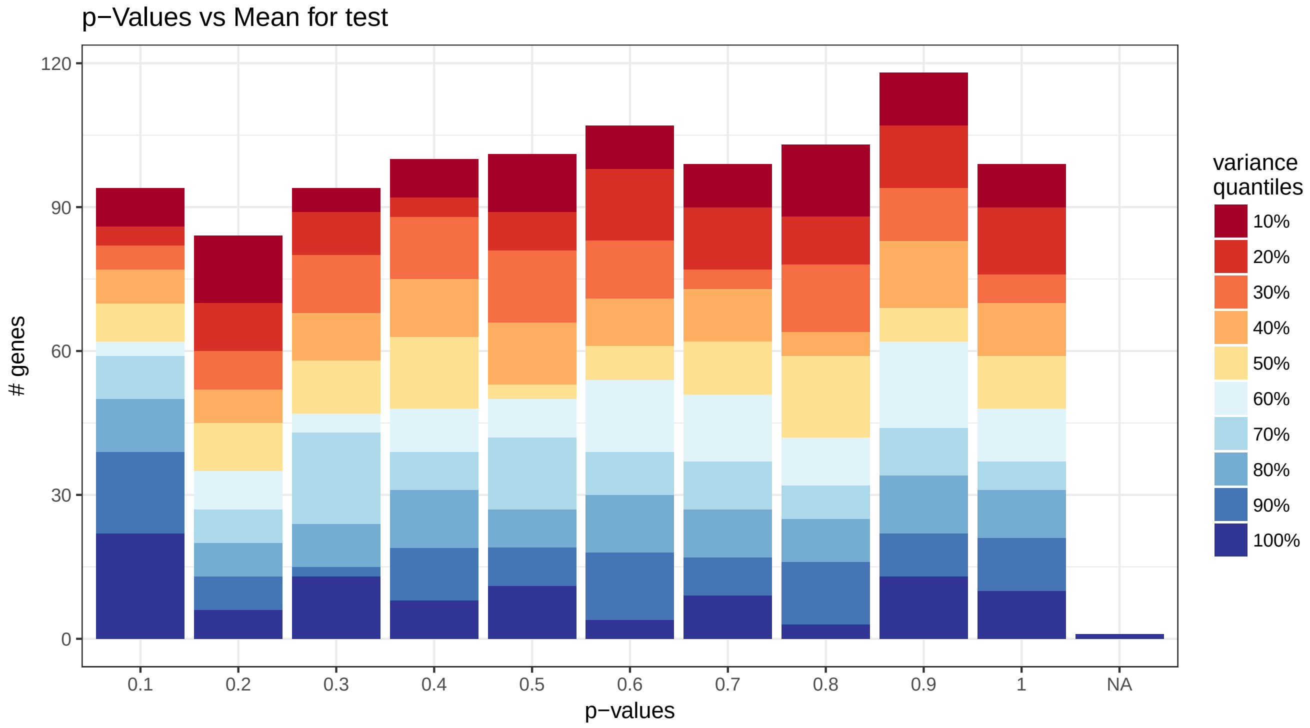
```
## ## Comparison:  test {.tabset}
##
##
##  <br>out of 1000 with nonzero total read count<br>adjusted p-value < 0.1<br>LFC > 0 (up)     : 6, 0.6%
##
##
## Differential expression file at:  test_de.csv
##
## Normalized counts matrix file at:  test_log2_counts.csv
##
## ### MA plot plot
```



```
##
##
## ### Volcano plot
```

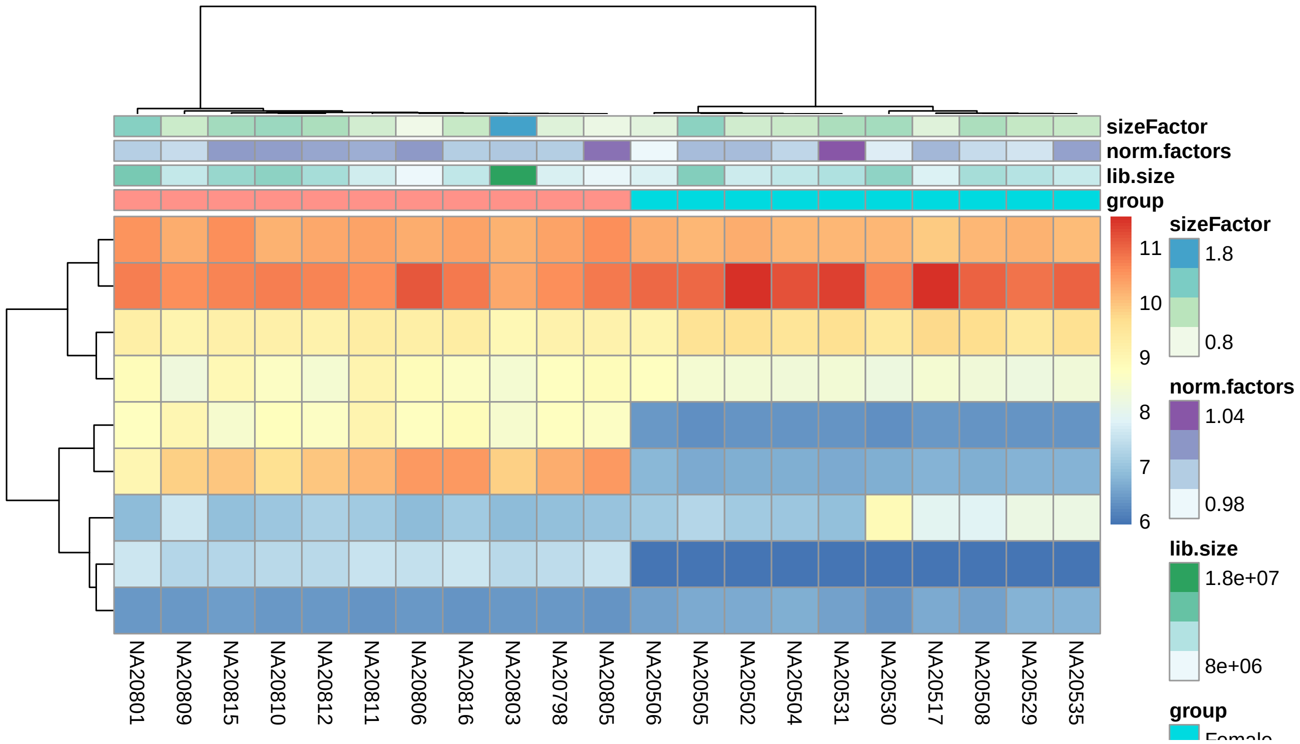


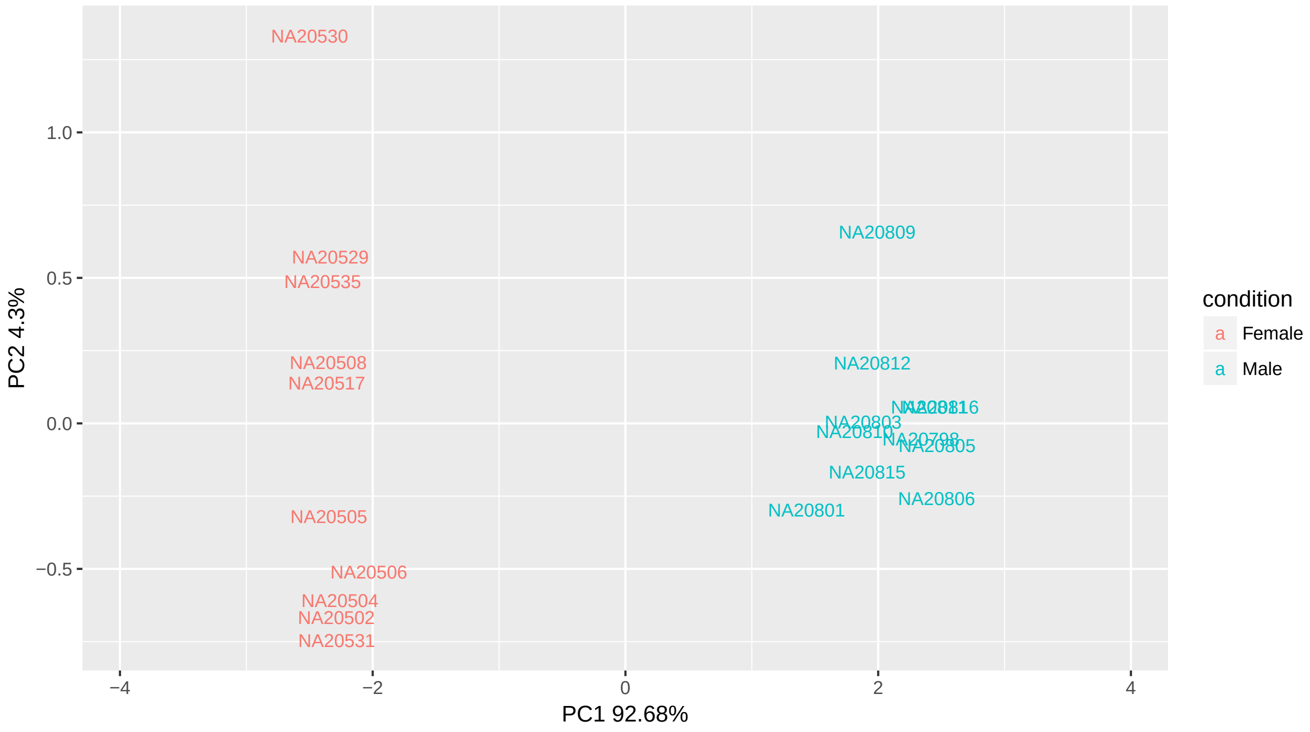
```
##
##
## ### QC for DE genes p-values/variance
```



```
##
##
## ### Most significand, FDR< 0.05  and log2FC >  0.1 :  9
```

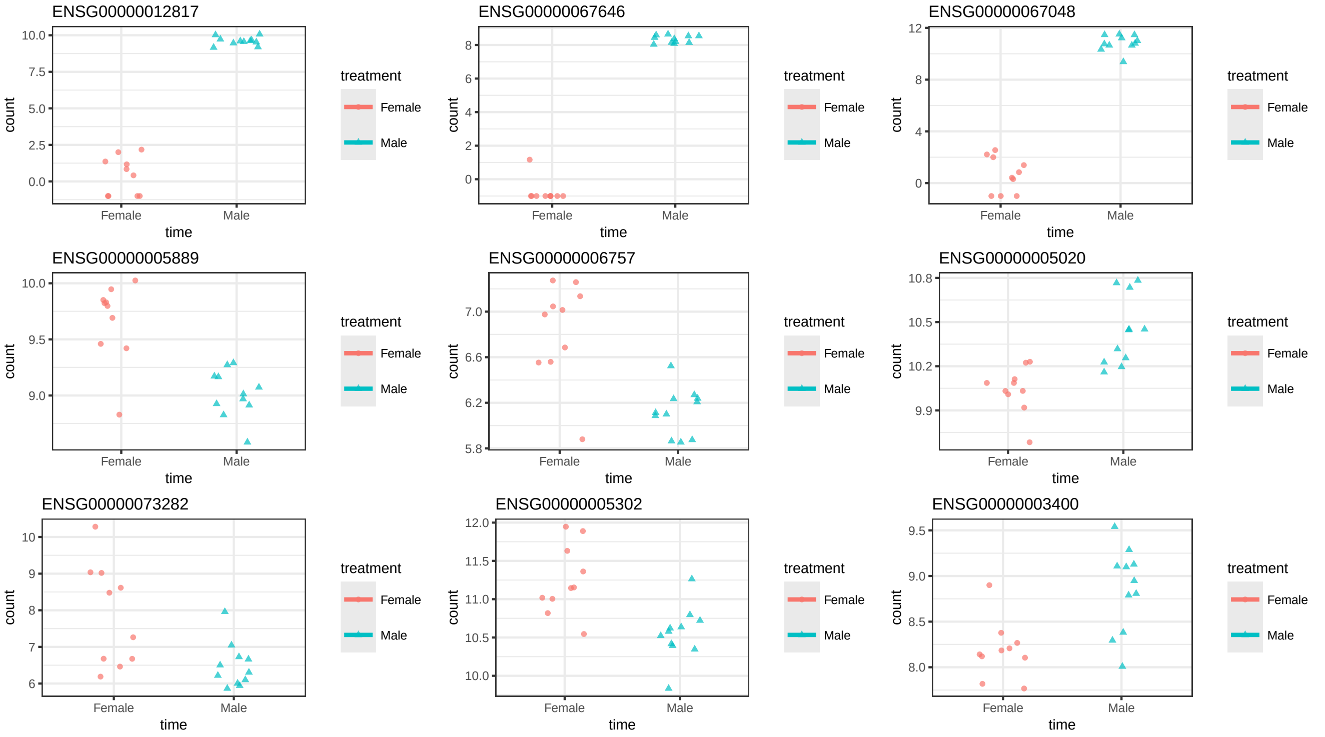
```
##
##
## ### Plots most significand
```





```
##
##
```

```
##
## Plot top 9 genes
```



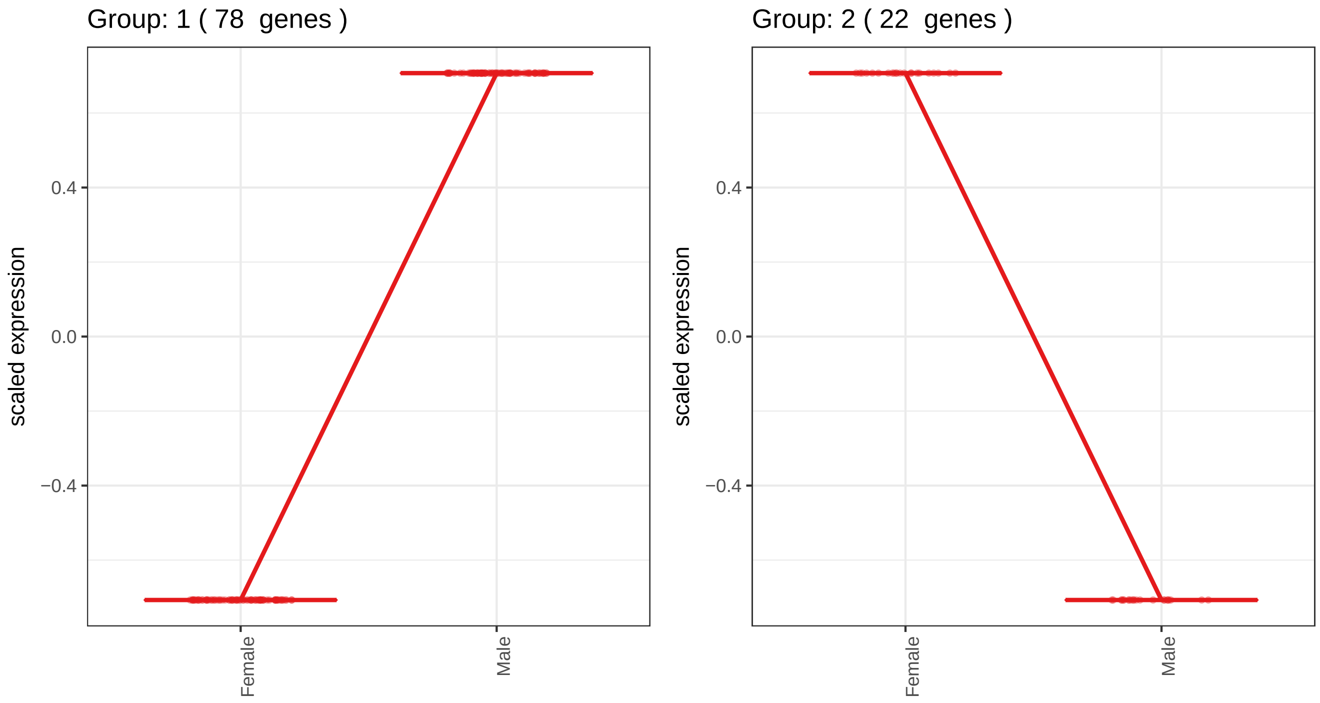
```
##
##
##
## ### Top DE genes
##
##
## \begin{tabular}{l|r|r|r|r|r|r|r}
## \hline
##   & baseMean & log2FoldChange & lfcSE & stat & pvalue & padj & absMaxLog2FC\\
## \hline
## ENSG00000012817 & 414.71514 & 5.0383666 & 0.1280628 & 39.342935 & 0.0000000 & 0.0000000 & 5.0383666\\
## \hline
## ENSG00000067646 & 171.03009 & 4.4844515 & 0.1304760 & 34.369939 & 0.0000000 & 0.0000000 & 4.4844515\\
## \hline
## ENSG00000067048 & 1032.64158 & 3.6088039 & 0.1418177 & 25.446777 & 0.0000000 & 0.0000000 & 3.6088039\\
## \hline
## ENSG00000005889 & 670.30349 & -0.5654992 & 0.1050602 & -5.382623 & 0.0000001 & 0.0000184 & 0.5654992\\
## \hline
## ENSG00000006757 & 92.82614 & -0.5687093 & 0.1228764 & -4.628304 & 0.0000037 & 0.0007373 & 0.5687093\\
## \hline
## ENSG00000005020 & 1238.12238 & 0.3638754 & 0.0855898 & 4.251386 & 0.0000212 & 0.0030350 & 0.3638754\\
## \hline
## ENSG00000073282 & 220.50620 & -0.5703537 & 0.1336781 & -4.266619 & 0.0000198 & 0.0030350 & 0.5703537\\
## \hline
## ENSG00000005302 & 2024.85620 & -0.5313921 & 0.1267987 & -4.190831 & 0.0000278 & 0.0034742 & 0.5313921\\
## \hline
## ENSG00000003400 & 394.82940 & 0.5090003 & 0.1273944 & 3.995469 & 0.0000646 & 0.0071740 & 0.5090003\\
## \hline
## ENSG00000069702 & 106.95162 & -0.4442940 & 0.1291537 & -3.440042 & 0.0005816 & 0.0581623 & 0.4442940\\
## \hline
## ENSG00000010278 & 84.64206 & 0.4736610 & 0.1406137 & 3.368527 & 0.0007557 & 0.0687009 & 0.4736610\\
## \hline
## ENSG00000023171 & 165.99513 & -0.4299847 & 0.1332187 & -3.227661 & 0.0012481 & 0.0999395 & 0.4299847\\
## \hline
## ENSG00000072501 & 3694.13874 & -0.4022772 & 0.1250802 & -3.216153 & 0.0012992 & 0.0999395 & 0.4022772\\
## \hline
## ENSG00000059377 & 132.55552 & 0.4494421 & 0.1435071 & 3.131846 & 0.0017371 & 0.1240792 & 0.4494421\\
## \hline
## ENSG00000068079 & 1039.04270 & 0.3447312 & 0.1138155 & 3.028860 & 0.0024548 & 0.1534241 & 0.3447312\\
## \hline
## ENSG00000070018 & 119.93653 & -0.4286051 & 0.1408198 & -3.043642 & 0.0023373 & 0.1534241 & 0.4286051\\
## \hline
## ENSG00000008277 & 378.03916 & -0.4001595 & 0.1376207 & -2.907699 & 0.0036410 & 0.2022771 & 0.4001595\\
## \hline
## ENSG00000012963 & 1836.75128 & 0.2397646 & 0.0821682 & 2.917972 & 0.0035232 & 0.2022771 & 0.2397646\\
## \hline
## ENSG00000005059 & 481.54751 & 0.3411158 & 0.1192221 & 2.861179 & 0.0042207 & 0.2221415 & 0.3411158\\
## \hline
## ENSG00000038427 & 100.44612 & -0.3833305 & 0.1384660 & -2.768408 & 0.0056331 & 0.2682422 & 0.3833305\\
## \hline
## \end{tabular}
```

3 Detect patterns of expression

In this section, we show how to detect pattern of expression. Mainly useful when data is a time course experiment. `degPatterns` needs a expression matrix, the design experiment and the column used to group samples.

```r
data(humanSexDEedgeR)
ma <- humanSexDEedgeR$counts[1:100,]
des <- data.frame(row.names=colnames(ma),
sex=as.factor(humanSexDEedgeR$samples$group))
res <- degPatterns(ma, des, time="sex", col=NULL)
```

```
##
##
## Working with  100  genes
##
##
##
##  Working with  100 genes after filtering
```


Group: 1 (78 genes)
scaled expression
0.4
0.0
−0.4
Female
Male
Group: 2 (22 genes)
scaled expression
0.4
0.0
−0.4
Female
Male