DEGreport

Lorena Pantano

Modified: 23 May, 2014. Compiled: May 15, 2016

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

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)
```

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 following lines to create the report. You will need Nozzle.R1 package.

```
pathreport<-"~/report" #change this to a proper path
createReport(g1,g2,counts,detag10,pval,fc,pathreport,colors,pop=400)
```

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

```
degObj(counts,design,"/tmp/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(g1,g2,pval,counts)
degMB(detags,g1,g2,counts)
degVB(detags,g1,g2,counts)
```

To create a ranked list of genes to validate, you can use degRank function. It will use the variation of each genes to give the ones with highest FC not influenced by the variation.

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