pcaGoPromoter version 1.16.0

Morten Hansen

May 3, 2016

1 Introduction

This R package provides functions to ease the analysis of Affymetrix DNA micro arrays by principal component analysis with annotation by GO terms and possible transcription factors.

2 Requirements

R version 2.14.0 or higher

```
> source("http://bioconductor.org/biocLite.R")
> biocLite("pcaGoPromoter",dependencies=TRUE)
```

Rgraphviz from Bioconductor is needed to draw Gene Ontology tree. Note: Graphviz needs to be installed on the computer for Rgraphviz to work. See Rgraphviz README for installation.

3 Example

3.1 Load the library

```
> library("pcaGoPromoter")
```

3.2 Read in data set serumStimulation

```
> library("serumStimulation")
> data(serumStimulation)
```

The serumStimulation data set has been created from 13 CEL files - 5 controls, 5 serum stimulated with inhibitor and 3 serum stimulated without inhibitor. They are read with ReadAffy(), normalized with rma() and the expression data extracted with exprs(). All of these function are part of the affy package.

The arrays are most likely grouped in some sort of way. Create a factor vector to indicate the groups:

```
> groups <- as.factor( c( rep("control",5) , rep("serumInhib",5) ,
+                         rep("serumOnly",3) ) )
> groups
```

3.3 Make PCA informative plot

This function "does-it-all". It will make a PCA plot and annotate the axis will GO terms and possible commmon transcription factors.

```
> pcaInfoPlot(eData=serumStimulation,groups=groups)
```

3.4 Principal component analysis (PCA)

```
> pcaOutput <- pca(serumStimulation)
```

```
> plot(pcaOutput, groups=groups)
```

Proportion of variance is noted along the axis. In this case there are 3 groups in the data set - control, serumInhib and serumOnly. There is a clear separation of the groups along the 1. principal component (X-axis). The 2. principal component shown a difference between the controls and the serum stimulated.

3.5 Get loadings from PCA

We would like to have the first 1365 probe ids (2,5 %) from 2. principal component in the negative (serum stimulated) direction.

```
> loadsNegPC2 <- getRankedProbeIds( pcaOutput, pc=2, decreasing=FALSE )[1:1365]
```

3.6 Create Gene Ontology tree from loadings

Note: In this step you will be asked to install the necessary data packages.

```
> GOtreeOutput <- GOtree( input = loadsNegPC2)
```

```
> plot(GOtreeOutput,legendPosition = "bottomright")
```

Output to PDF file is advised. This can be done by coping output to a PDF file:

```
> dev.copy2pdf(file="GOtree.pdf")
```

Function 'GOtree()' also outputs a list of GO terms order by p-value.

```
> head(GOtreeOutput$sigGOs,n=10)
```

3.7 Get list of possible transcription factors

To get possible transcription factors, use function primo() function.

```
> TFtable <- primo( loadsNegPC2 )
> head(TFtable$overRepresented)
```

The output shows you which possible transcription factors (genes) the supplied probes have in common.

3.8 Get a list of probe ids for a specific transcription factor

```
> probeIds <- primoHits( loadsNegPC2 , id = 9343 )
> head(probeIds)
```