

pcaGoPromoter version 1.11.0

Morten Hansen

October 13, 2014

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

[1] control    control    control    control    control    serumInhib
[7] serumInhib serumInhib serumInhib serumInhib serumOnly  serumOnly
[13] serumOnly
Levels: control serumInhib serumOnly

```

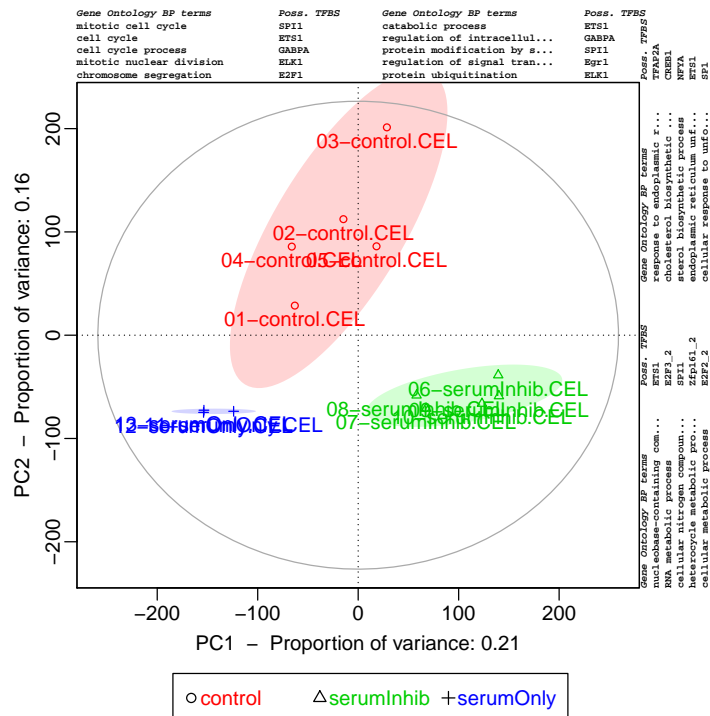
3.3 Make PCA informative plot

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

```

> pcaInfoPlot(serumStimulation,groups=groups)

```



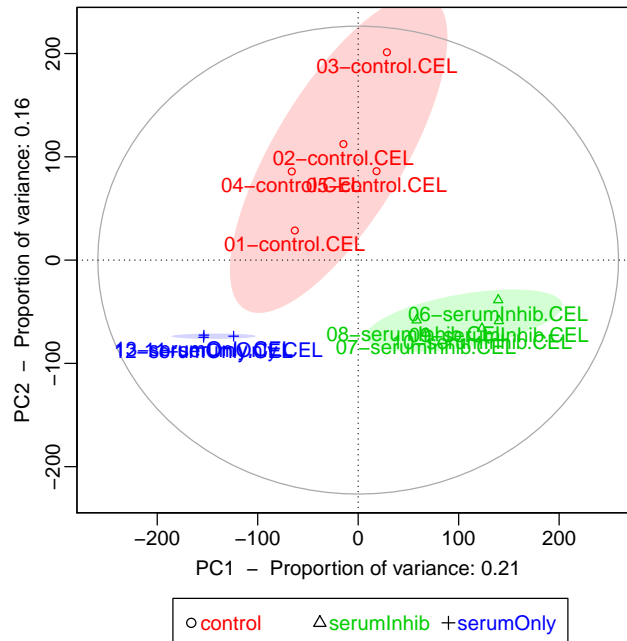
3.4 Principal component analysis (PCA)

```

> pcaOutput <- pca(serumStimulation)
> plot(pcaOutput, groups=groups)

```

PCA plot of 1. and 2. principal component



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.

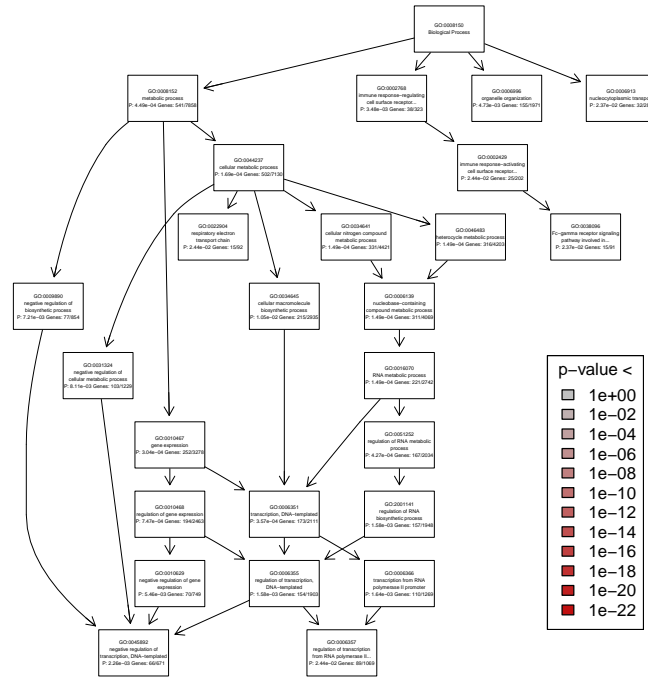
```
> loadsNegPC2 <- getRankedProbeIds( pcaOutput, pc=2, decreasing=FALSE )[1:1365]
```

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")
```

Gene Ontology tree, biological processes



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

```
> dev.copy2pdf(file="G0tree.pdf")
```

Function 'G0tree()' also outputs a list of GO terms order by p-value.

```
> head(G0treeOutput$sigGOs,n=10)
```

	GOid	genesInTerm	totalGenesInTerm	pValue	GOterm
287	GO:0006139	311	4069	0.000148921	nucleobase-containing compound metabolic process
1012	GO:0016070	221	2742	0.000148921	RNA metabolic process
1597	GO:0034641	331	4421	0.000148921	cellular nitrogen compound metabolic process
2142	GO:0046483	316	4203	0.000148921	heterocycle metabolic process
1955	GO:0044237	502	7130	0.000169008	cellular metabolic process
862	GO:0010467	252	3278	0.000303639	gene expression
333	GO:0006351	173	2111	0.000356736	
2434	GO:0051252	167	2034	0.000426658	
723	GO:0008152	541	7858	0.000449290	
863	GO:0010468	194	2463	0.000747396	

```

333             transcription, DNA-templated
2434          regulation of RNA metabolic process
723             metabolic process
863             regulation of gene expression

```

3.7 Get list of possible transcription factors

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

```

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

      id baseId pwmLength  gene      pValue
1  9326 MA0098         6   ETS1 2.30355e-08
2 10235 PB0113        17  E2F3_2 1.08742e-07
3   9308 MA0080         6   SPI1 3.92539e-05
4 10321 PB0199        14 Zfp161_2 7.41396e-05
5 10234 PB0112        17  E2F2_2 9.72520e-05
6 10132 PB0010        14  Egr1_1 1.08150e-04

```

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)

[1] "NM_001121"      "NM_016824"      "NM_001114380"  "NM_002209"      "NM_003342"
[6] "NM_006403"

```