

pcaGoPromoter version 1.0.0

Morten Hansen

March 31, 2012

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:

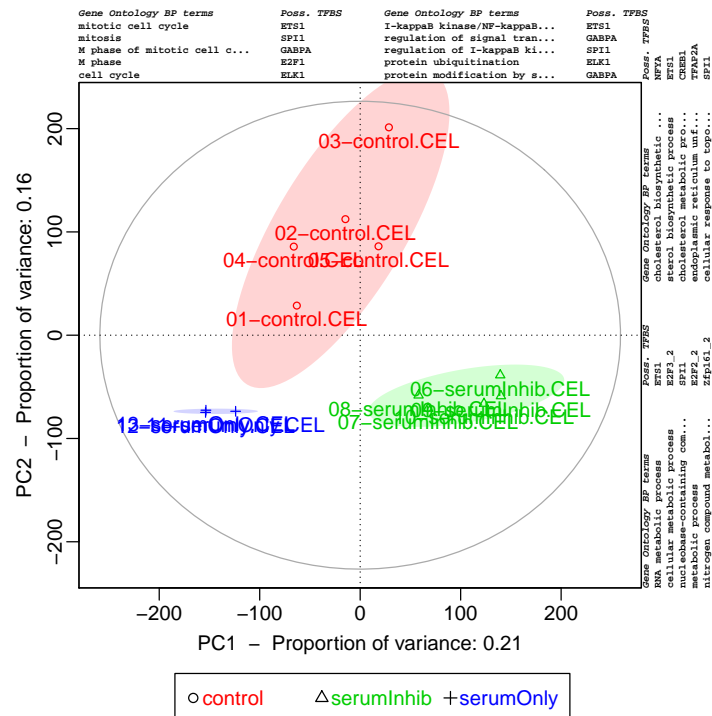
```
> groups <- as.factor( c( rep("control",5) , rep("serumInhib",5) ,
+                           rep("serumOnly",3) ) )
> groups

[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

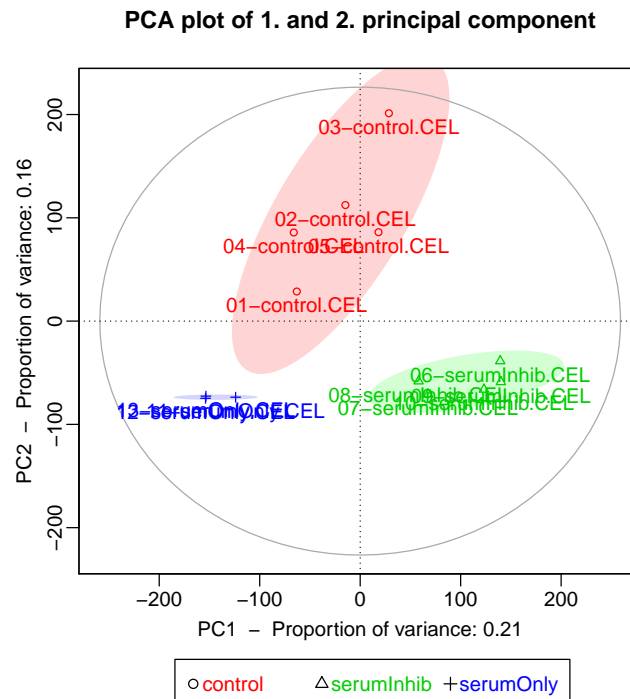
This function "does-it-all". It will make a PCA plot and annotate the axis with GO terms and possible common transcription factors.

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



3.4 Principal component analysis (PCA)

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



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.

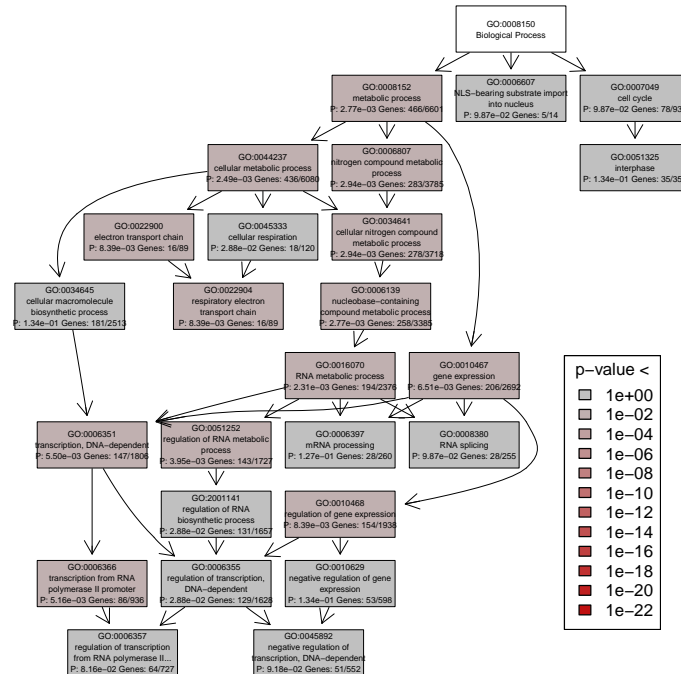
```
> loadsNegPC2 <- getRankedProbeIds( pcaOutput, pc=2, decreasing=FALSE )[1:1365]
```

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 coping 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
903	GO:0016070	194	2376	0.00230688
1651	GO:0044237	436	6080	0.00249049
246	GO:0006139	258	3385	0.00277233
656	GO:0008152	466	6601	0.00277233
424	GO:0006807	283	3785	0.00294415
1381	GO:0034641	278	3718	0.00294415
2050	GO:0051252	143	1727	0.00395398
296	GO:0006366	86	936	0.00516168
285	GO:0006351	147	1806	0.00550454
779	GO:0010467	206	2692	0.00651298
	GOterm			
903	RNA metabolic process			
1651	cellular metabolic process			
246	nucleobase-containing compound metabolic process			
656	metabolic process			
424	nitrogen compound metabolic process			
1381	cellular nitrogen compound metabolic process			

```

2050             regulation of RNA metabolic process
296      transcription from RNA polymerase II promoter
285             transcription, DNA-dependent
779             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	1.79524e-08
2	10235	PB0113	17	E2F3_2	3.53616e-08
3	9308	MA0080	6	SPI1	3.05400e-05
4	10234	PB0112	17	E2F2_2	7.77734e-05
5	10321	PB0199	14	Zfp161_2	2.12513e-04
6	10217	PB0095	16	Zfp161_1	2.78272e-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"				