

pcaGoPromoter version 1.2.0

Morten Hansen

October 1, 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:

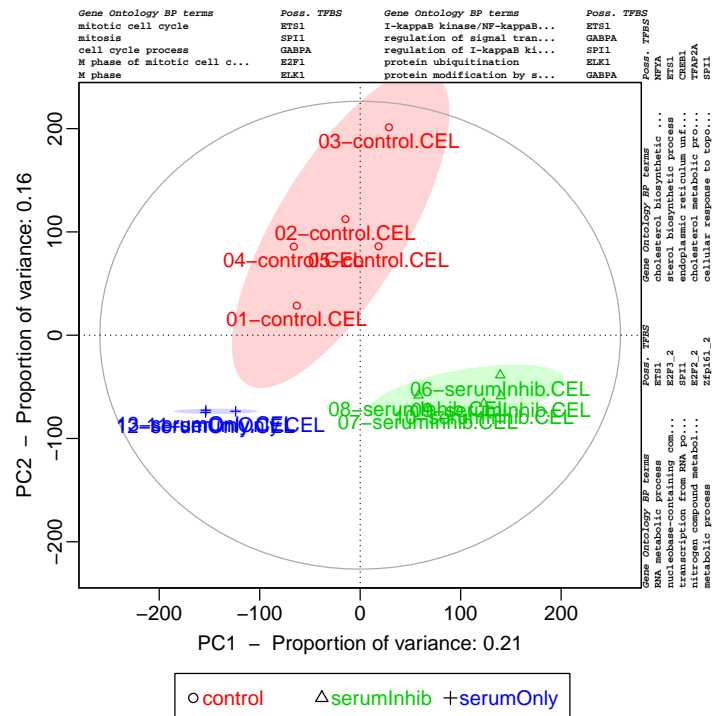
```
> groups <- as.factor( c( rep("control",5) , rep("serumInhib",5) ,
+                           rep("serumOnly",3) ) )
> groups

[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

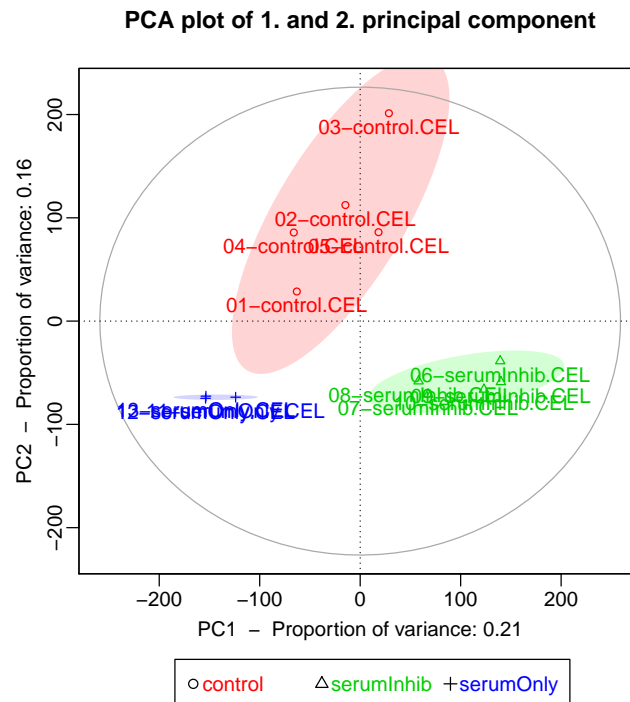
This function "does-it-all". It will make a PCA plot and annotate the axis with GO terms and possible common transcription factors.

```
> 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.

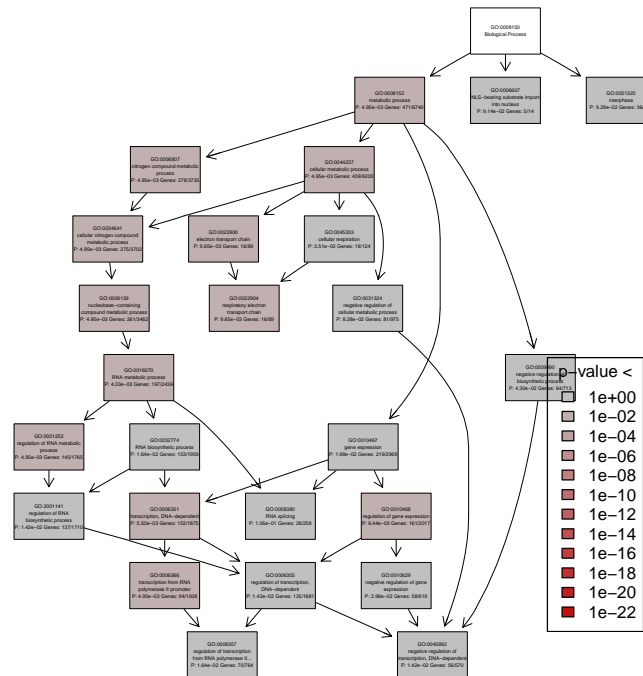
```
> loadsNegPC2 <- getRankedProbeIds( pcaOutput, pc=2, decreasing=FALSE )[1:1365]
```

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

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

	G0id	genesInTerm	totalGenesInTerm	pValue
922	G0:0016070	197	2439	0.00433166
257	G0:0006139	261	3462	0.00495039
308	G0:0006366	94	1008	0.00495039
437	G0:0006807	278	3735	0.00495039
670	G0:0008152	471	6740	0.00495039
1417	G0:0034641	275	3702	0.00495039
1707	G0:0044237	439	6203	0.00495039
2114	G0:0051252	145	1765	0.00495039
296	G0:0006351	152	1875	0.00550282
797	G0:0010468	161	2017	0.00644003
	G0term			
922	RNA metabolic process			
257	nucleobase-containing compound metabolic process			
308	transcription from RNA polymerase II promoter			
437	nitrogen compound metabolic process			
670	metabolic process			
1417	cellular nitrogen compound metabolic process			

```

1707                                cellular metabolic process
2114                    regulation of RNA metabolic process
296                                transcription, DNA-dependent
797                                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	2.52317e-08
2	10235	PB0113	17	E2F3_2	6.14410e-08
3	9308	MA0080	6	SPI1	4.74868e-05
4	10234	PB0112	17	E2F2_2	5.98521e-05
5	10321	PB0199	14	Zfp161_2	2.04492e-04
6	10132	PB0010	14	Egr1_1	4.59633e-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"				