R: Microarray

```
>
> v1<-1
> v2<-as.integer(2)
> v3<-TRUE
> v4<-"abc"
> typeof(v1)
[1] "double"
> typeof(v2)
[1] "integer"
> typeof(v3)
[1] "logical"
> typeof(v4)
[1] "character"
>
```

```
>
> c(1:10)
 [1]  1  2  3  4  5  6  7  8  9 10
> c(6,3,6,1:4,2:3)
[1] 6 3 6 1 2 3 4 2 3
> runif(10)
 [1] 0.15281516 0.53352757 0.30039152 0.69891784 0.35381320 0.81001772
 [7] 0.90448145 0.89338562 0.07840998 0.46073054
> x<-c(6,3,1,5,7,5)
> sort(x)
[1] 1 3 5 5 6 7
> sum(x)
[1] 27
> prod(x)
[1] 3150
> range(x)
[1] 1 7
> x<3
[1] FALSE FALSE  TRUE FALSE FALSE FALSE
> which(x<3)
[1] 3
> x[x<3]
[1] 1
> which(x<5)
[1] 2 3
> x(x<5)
錯誤: 沒有這個函數 "x"
> x[x<5]
[1] 3 1
>
```

```
> mode(x)
[1] "numeric"
> l1<-list(a=3,b=c(4:7),c=c("Abb","bbb","ddd"))
> l1
$a
[1] 3

$b
[1] 4 5 6 7

$c
[1] "Abb" "bbb" "ddd"

> mode(l1)
[1] "list"
> typeof(l1)
[1] "list"
> l1$a
[1] 3
> l1$b
[1] 4 5 6 7
>
```

```
>
> m1<-matrix(c(1,2,3,4,5,6),3,2)
> m1
     [,1] [,2]
[1,]    1    4
[2,]    2    5
[3,]    3    6
> colnames(m1)=c("c1","c2")
> m1
     c1 c2
[1,]  1  4
[2,]  2  5
[3,]  3  6
> rownames(m1)=c("r1","r2","r3")
> m1
   c1 c2
r1  1  4
r2  2  5
r3  3  6
>
```

```
> f1<-as.data.frame(m1)
> f1
   c1 c2
r1  1  4
r2  2  5
r3  3  6
> m1$c1
Error in m1$c1 : $ operator is invalid for atomic vectors
> f1$c1
[1] 1 2 3
> f1$aa<-c("x","y","z")
> f1
   c1 c2 aa
r1  1  4  x
r2  2  5  y
r3  3  6  z
> cbind(f1,c("tmpa","tmpb","tmpc"))
   c1 c2 aa c("tmpa", "tmpb", "tmpc")
r1  1  4  x                      tmpa
r2  2  5  y                      tmpb
r3  3  6  z                      tmpc
> cbind(m1,c("tmpa","tmpb","tmpc"))
   c1  c2
r1 "1" "4" "tmpa"
r2 "2" "5" "tmpb"
r3 "3" "6" "tmpc"
>
```

```
> colors<-as.factor(c("blue","red","red","blue","blue"))
> colors
[1] blue red  red  blue blue
Levels: blue red
> class(colors)
[1] "factor"
> typeof(colors)
[1] "integer"
> class(m1)
[1] "matrix"
> typeof(m1)
[1] "double"
> class(f1)
[1] "data.frame"
> typeof(f1)
[1] "list"
> class(l1)
[1] "list"
> typeof(l1)
[1] "list"
>
```

```
> testmatrix<-matrix(runif(36,10,20),6,6)
> testmatrix
         [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
[1,] 16.13157 18.91963 10.91774 10.17362 10.49850 18.95450
[2,] 19.77957 17.26479 12.27047 11.23420 18.98668 12.25177
[3,] 11.45185 14.15188 13.43486 17.70140 15.47825 17.42012
[4,] 19.24737 16.83449 15.43159 10.57259 13.51745 11.64292
[5,] 13.36099 12.20518 18.26361 15.86578 19.57646 15.34650
[6,] 16.88365 11.86373 18.82184 17.94149 13.16569 16.73511
> testmatrix2<-matrix(runif(12),6,2)
> testmatrix2
           [,1]       [,2]
[1,] 0.23453831 0.62195731
[2,] 0.62870788 0.04836163
[3,] 0.96935439 0.18614407
[4,] 0.87793829 0.79510670
[5,] 0.09370572 0.11846387
[6,] 0.94481782 0.97245266
> testmatrix %*% testm
testmatrix2                          testmatrix
> testmatrix %*% testmatrix2
         [,1]     [,2]
[1,] 54.08568 40.74557
[2,] 50.60580 38.51697
[3,] 58.05641 43.15614
[4,] 51.60608 36.98754
[5,] 58.77432 42.15773
[6,] 62.46057 46.67739
>
```

```
> testmatrix + 1
         [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
[1,] 17.13157 19.91963 11.91774 11.17362 11.49850 19.95450
[2,] 20.77957 18.26479 13.27047 12.23420 19.98668 13.25177
[3,] 12.45185 15.15188 14.43486 18.70140 16.47825 18.42012
[4,] 20.24737 17.83449 16.43159 11.57259 14.51745 12.64292
[5,] 14.36099 13.20518 19.26361 16.86578 20.57646 16.34650
[6,] 17.88365 12.86373 19.82184 18.94149 14.16569 17.73511
> testmatrix * 2
         [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
[1,] 32.26315 37.83926 21.83549 20.34724 20.99700 37.90900
[2,] 39.55914 34.52958 24.54095 22.46841 37.97336 24.50354
[3,] 22.90371 28.30377 26.86973 35.40279 30.95651 34.84023
[4,] 38.49474 33.66898 30.86318 21.14518 27.03491 23.28583
[5,] 26.72197 24.41036 36.52722 31.73157 39.15291 30.69300
[6,] 33.76730 23.72746 37.64369 35.88298 26.33139 33.47021
> testmatrix %*% solve(testmatrix)
              [,1]          [,2]          [,3]          [,4]          [,5]
[1,]  1.000000e+00 -2.207436e-16  6.635317e-17  1.142749e-16  5.659535e-17
[2,]  2.710505e-18  1.000000e+00 -8.044780e-17 -3.426079e-17  6.142005e-17
[3,] -1.131907e-16 -2.921925e-16  1.000000e+00  2.578233e-16 -5.529431e-17
[4,] -8.109832e-17 -2.013906e-16 -1.075529e-16  1.000000e+00  2.107689e-16
[5,] -5.052382e-17 -1.830675e-16  2.225867e-16  8.044780e-17  1.000000e+00
[6,] -2.623769e-17 -4.237604e-16 -2.255141e-17  1.589440e-16  1.117812e-16
              [,6]
[1,] -3.183218e-16
[2,] -1.432773e-16
[3,] -2.773389e-16
[4,] -2.668222e-16
[5,] -1.401873e-16
[6,]  1.000000e+00
> t(testmatrix)
         [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
[1,] 16.13157 19.77957 11.45185 19.24737 13.36099 16.88365
[2,] 18.91963 17.26479 14.15188 16.83449 12.20518 11.86373
[3,] 10.91774 12.27047 13.43486 15.43159 18.26361 18.82184
[4,] 10.17362 11.23420 17.70140 10.57259 15.86578 17.94149
[5,] 10.49850 18.98668 15.47825 13.51745 19.57646 13.16569
[6,] 18.95450 12.25177 17.42012 11.64292 15.34650 16.73511
>
```

File Edit View Workspace Run Data Analysis Plots Distributions Windows Settings Help

Open Create Save Run selection Interrupt running command

Workspace Files

All Non-Functions Functions

Show All Environments

Show Hidden Objects

Name	Label	Type	Class
package:base			
Autoloads			
package:methods			
package:datasets			
package:utils			
package:grDevices			
package:graphics			
package:stats			
package:rkward			
.GlobalEnv			
mycmap3			data.fra
mycmap2			data.fra
mycmap			data.fra
hg			data.fra
cmap2			data.fra
cmap			data.fra
columnna...		Factor	factor
Vendor		Factor	factor
Vehicle		Factor	factor
Scanner		Factor	factor
Sample		Number	integer
Concentra...		Number	numeri
Cmap		Factor	factor
Cell		Factor	factor
Batch		Factor	factor
Array.1		Factor	factor

Update

rkward_welcome | read.table.html | read.table.html | data.frame.html | cmap

	1	2	3	4	5	6	7	8	9	10	#New Variable#
Name	**Sample**	**Cmap**	**Concentrati...**	**Cell**	**Array.1**	**Scanner**	**Vehicle**	**Vendor**	**Batch**	**columnname**	
Label											
Type	Number	Factor	Number	Factor	Factor	Factor	Factor	Factor	Factor	Factor	
Format											
Levels		0173570-00...		HL60#,#MC...	HG-U133A#,...	Axon ImageX...	DMSO#,#eth...	Asinex#,#As...	1#,#1000#,...	s1#,#s1000...	
1	1	metformin	1e-05	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	1	s1	
2	2	metformin	1e-05	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	1	s2	
3	3	metformin	1e-07	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	1	s3	
4	4	metformin	0.001	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	1	s4	
21	21	phenformin	1e-05	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	2	s21	
22	22	phenyl bigua...	1e-05	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	2	s22	
23	23	valproic acid	0.001	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	2	s23	
61	61	metformin	1e-05	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	2	s61	
121	121	estradiol	1e-08	MCF7	HG-U133A	HP GeneArra...	DMSO	Sigma-Aldrich	5	s121	
122	122	alpha-estradiol	1e-08	MCF7	HG-U133A	HP GeneArra...	DMSO	Sigma-Aldrich	5	s122	
123	123	dexamethas...	1e-06	MCF7	HG-U133A	HP GeneArra...	DMSO	Sigma-Aldrich	5	s123	
124	124	mesalazine	0.0001	MCF7	HG-U133A	HP GeneArra...	DMSO	Sigma-Aldrich	5	s124	
141	141	chlorpropami...	0.0001	MCF7	HG-U133A	HP GeneArra...	DMSO	Sigma-Aldrich	6	s141	

```
>
> cmap<-read.delim("biodata/cmap/pvca-rscript/b2cmap_instance.csv",header=TRUE,sep="\t",row.names=1)
> dim(cmap)
[1] 6092   10
> cmap[1,5]
[1] HG-U133A
Levels: HG-U133A HT_HG-U133A HT_HG-U133A_EA
> cmap[5,1]
[1] 21
> mycmap<-cmap[,c(2:3,9)]
> dim(mycmap)
[1] 6092    3
> mycmap2<-cmap[c(3:6),]
> dim(mycmap2)
[1]  4 10
> mycmap3<-cmap[c(3:6),c(2:3,9)]
> dim(mycmap3)
[1] 4 3
>
```

	1	2	3
Name	**Cmap**	**Concentrati...**	**Batch**
Label			
Type	Factor	Number	Factor
Format			
Levels	0173570-00...		1#,#1000#,...
1	metformin	1e-05	1
2	metformin	1e-05	1
3	metformin	1e-07	1
4	metformin	0.001	1
21	phenformin	1e-05	2
22	phenyl bigua...	1e-05	2
23	valproic acid	0.001	2
61	metformin	1e-05	2
121	estradiol	1e-08	5
122	alpha-estradiol	1e-08	5
123	dexamethas...	1e-06	5
124	mesalazine	0.0001	5
141	chlorpropami...	0.0001	6

	1	2	3	4	5	6	7	8	9	10	#New Variable#
Name	**Sample**	**Cmap**	**Concentrati...**	**Cell**	**Array.1**	**Scanner**	**Vehicle**	**Vendor**	**Batch**	**columnname**	
Label											
Type	Number	Factor	Number	Factor	Factor	Factor	Factor	Factor	Factor	Factor	
Format											
Levels		0173570-00...		HL60#,#MC...	HG-U133A#,...	Axon ImageX...	DMSO#,#eth...	Asinex#,#As...	1#,#1000#,...	s1#,#s1000...	
3	3	metformin	1e-07	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	1	s3	
4	4	metformin	0.001	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	1	s4	
21	21	phenformin	1e-05	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	2	s21	
22	22	phenyl bigua...	1e-05	MCF7	HG-U133A	HP GeneArra...	medium	Sigma-Aldrich	2	s22	

	1	2	3	#New Variable#
Name	**Cmap**	**Concentrati...**	**Batch**	
Label				
Type	Factor	Number	Factor	
Format				
Levels	0173570-00...		1#,#1000#,...	
3	metformin	1e-07	1	
4	metformin	0.001	1	
21	phenformin	1e-05	2	
22	phenyl bigua...	1e-05	2	

[Unnamed Workspace] – mycm

File Edit View Workspace Run Data Analysis Plots Distributions Windows Settings Help

Open Create Save Run selection Interrupt running command

Workspace Files

All Non-Functions Functions

Show All Environments

Show Hidden Objects

Name	Label	Type	Class
package:base			
Autoloads			
package:methods			
package:datasets			
package:utils			
package:grDevices			
package:graphics			
package:stats			
package:rkward			
.GlobalEnv			
mycmap4			data.fra
Concentra...		Number	numeri
Cmap		Factor	factor
Batch		Factor	factor
mycmap3			data.fra
mycmap2			data.fra
mycmap			data.fra
Concentra...		Number	numeri
Cmap		Factor	factor
Batch		Factor	factor
hg			data.fra
cmap2			data.fra
cmap			data.fra

Update

rkward_welcome | read.table.html | read.table.html | data.frame

	1	2	3	#New Variable#
Name	**Cmap**	**Concentrati...**	**Batch**	
Label				
Type	Factor	Number	Factor	
Format				
Levels	0173570-00...		1#,#1000#,...	
1	metformin	1e-05	1	
2	metformin	1e-05	1	
3	metformin	1e-07	1	
4	metformin	0.001	1	
61	metformin	1e-05	2	
1694	metformin	2.42e-05	627	
1816	metformin	2.42e-05	628	
1858	metformin	2.42e-05	629	
5068	metformin	2.42e-05	718	
5487	metformin	2.42e-05	737	

```
>
> cmap[["Cmap"]][4]
[1] metformin
1309 Levels: 0173570-0000 0175029-0000 0179445-0000 0198306-0000 ... zuclopenthixol
> cmap[["Cmap"]][5]
[1] phenformin
1309 Levels: 0173570-0000 0175029-0000 0179445-0000 0198306-0000 ... zuclopenthixol
> which(cmap[["Cmap"]]=="metformin")
 [1]    1    2    3    4    8  925 1046 1088 3930 4249
> mycmap4<-cmap[cmap[["Cmap"]]=="metformin",c(2,3,9)]
> mycmap4[,c("Cmap","Batch")]
          Cmap Batch
1    metformin     1
2    metformin     1
3    metformin     1
4    metformin     1
61   metformin     2
1694 metformin   627
1816 metformin   628
1858 metformin   629
5068 metformin   718
5487 metformin   737
>
```

```
> unique(cmap$Cell)
[1] MCF7   HL60   ssMCF7 PC3    SKMEL5
Levels: HL60 MCF7 PC3 SKMEL5 ssMCF7
> c1<-unique(cmap[cmap$Cell=="MCF7",2])
> c2<-unique(cmap[cmap$Cell=="HL60",2])
> c3<-unique(cmap[cmap$Cell=="ssMCF7",2])
> c4<-unique(cmap[cmap$Cell=="PC3",2])
> c5<-unique(cmap[cmap$Cell=="SKMEL5",2])
> length(c1)
[1] 1294
> length(c2)
[1] 1078
> length(c3)
[1] 16
> length(c4)
[1] 1182
> length(c5)
[1] 16
> length(unique(cmap$C
cmap[["Concentration"]]            cmap[["Cmap"]]                  cmap[["Cell"]]
> length(unique(cmap[["Cmap"]]))
[1] 1309
>
```

File Edit View Workspace Run Data Analysis Plots Distributions Windows Settings Help

Open Create Save Run selection Interrupt running command

Workspace Files

All Non-Functions Functions

Show All Environments

Show Hidden Objects

Name	Label	Type
package:base		
Autoloads		
package:methods		
package:datasets		
package:utils		
package:grDevices		
package:graphics		
package:stats		
package:rkward		
package:Biobase		
package:affy		
package:AnnotationDbi		
package:hgu133acdf		
.GlobalEnv		
x		
X6146151114...		Number
X6146151114...		Factor
X6146151114...		Number

Update

rkward_welcome | x

	1	2	3	4	5	6
Name	**X61461511...**	**X61461511...**	**X61461511...**	**X61461511...**	**X61461511...**	**X61461511...**
Label						
Type	Number	Factor	Number	Number	Factor	Number
Format						
Levels		A#,#M#,#P			A#,#M#,#P	
1007_s_a	7.75619582...	A	7.76175916...	0.23455651...	A	7.63209988...
1053_at	8.04026676...	P	7.60805950...	0.00080466...	P	8.18401420...
117_at	6.31622052...	P	6.45869139...	0.03133563...	P	6.33528244...
121_at	8.11570091...	P	8.05893416...	0.01309178...	P	8.17648677...
1255_g_	4.43889246...	A	4.44377345...	0.26746255...	A	4.44060560...
1294_at	6.91313351...	P	6.80276038...	0.01493651...	P	6.91510208...

```
> library(affy)
> list.celfiles(path="~/biodata/cmap/",full.names=TRUE)
[1] "/home/john/biodata/cmap//614615111406.A02.CEL"
[2] "/home/john/biodata/cmap//614615111406.A03.CEL"
[3] "/home/john/biodata/cmap//614615111406.A04.CEL"
[4] "/home/john/biodata/cmap//614615111406.A05.CEL"
[5] "/home/john/biodata/cmap//614615111406.A12.CEL"
> mydata<-ReadAffy(filenames=list.celfiles(path="~/biodata/cmap/",full.names=TRUE))
> eset <- rma(mydata)
Background correcting
Normalizing
Calculating Expression
> eset_PMA <- mas5calls(mydata)
Getting probe level data...
Computing p-values
Making P/M/A Calls
> x <- data.frame(exprs(eset), exprs(eset_PMA), assayDataElement(eset_PMA, "se.exprs"))
> x <- x[,sort(names(x))]
>
```

```
> eset
ExpressionSet (storageMode: lockedEnvironment)
assayData: 22283 features, 5 samples
  element names: exprs
protocolData
  sampleNames: 614615111406.A02.CEL 614615111406.A03.CEL ...
    614615111406.A12.CEL (5 total)
  varLabels: ScanDate
  varMetadata: labelDescription
phenoData
  sampleNames: 614615111406.A02.CEL 614615111406.A03.CEL ...
    614615111406.A12.CEL (5 total)
  varLabels: sample
  varMetadata: labelDescription
featureData: none
experimentData: use 'experimentData(object)'
Annotation: hgu133a
> pData(eset)
                     sample
614615111406.A02.CEL      1
614615111406.A03.CEL      2
614615111406.A04.CEL      3
614615111406.A05.CEL      4
614615111406.A12.CEL      5
> mydata
AffyBatch object
size of arrays=712x712 features (11 kb)
cdf=HG-U133A (22283 affyids)
number of samples=5
number of genes=22283
annotation=hgu133a
notes=
>
```

hgu133a.db

Affymetrix Human Genome U133 Set annotation data (chip hgu133a)

Bioconductor version: Release (2.10)

Affymetrix Human Genome U133 Set annotation data (chip hgu133a) assembled using data from public repositories

Author: Marc Carlson, Seth Falcon, Herve Pages, Nianhua Li

Maintainer: Biocore Data Team <biocannotation at lists.fhcrc.org>

To install this package, start R and enter:

```
source("http://bioconductor.org/biocLite.R")
biocLite("hgu133a.db")
```

To cite this package in a publication, start R and enter:

```
citation("hgu133a.db")
```

All Non-Functions Functions

Show All Environments

Show Hidden Objects

Name	Label	Type
x		
probleGene		
probeset		String
probeGene		
mydata		Unknc
mapCdfName		
eset_PMA		Unknc
eset		Unknc
SYMBOL		String

Update

rkward_welcome x Documentation for package 'hgu133a.db'

hgu133a.db	Bioconductor annotation data package
hgu133aACCNUM	Map Manufacturer identifiers to Accession Numbers
hgu133aALIAS2PROBE	Map between Common Gene Symbol Identifiers and Manufacturer Identifiers
hgu133aCHR	Map Manufacturer IDs to Chromosomes
hgu133aCHRLENGTHS	A named vector for the length of each of the chromosomes
hgu133aCHRLOC	Map Manufacturer IDs to Chromosomal Location
hgu133aENSEMBL	Map Ensembl gene accession numbers with Entrez Gene identifiers
hgu133aENTREZID	Map between Manufacturer Identifiers and Entrez Gene
hgu133aENZYME	Map between Manufacturer IDs and Enzyme Commission (EC) Numbers
hgu133aENZYME2PROBE	Map between Enzyme Commission Numbers and Manufacturer Identifiers
hgu133aGENENAME	Map between Manufacturer IDs and Genes
hgu133aGO	Map between Manufacturer IDs and Gene Ontology (GO)
hgu133aGO2ALLPROBES	Map between Gene Ontology (GO) Identifiers and all Manufacturer Identifiers in the subtree
hgu133aGO2PROBE	Map between Gene Ontology (GO) and Manufacturer Identifiers

```
>
>
>
>
> library(hgu133a.db)
> library(help=hgu133a.db)
> hgu133a()
Quality control information for hgu133a:


This package has the following mappings:

hgu133aACCNUM has 22283 mapped keys (of 22283 keys)
hgu133aALIAS2PROBE has 54726 mapped keys (of 110701 keys)
hgu133aCHR has 20380 mapped keys (of 22283 keys)
hgu133aCHRLENGTHS has 93 mapped keys (of 93 keys)
hgu133aCHRLOC has 20163 mapped keys (of 22283 keys)
hgu133aCHRLOCEND has 20163 mapped keys (of 22283 keys)
hgu133aENSEMBL has 19792 mapped keys (of 22283 keys)
hgu133aENSEMBL2PROBE has 13125 mapped keys (of 20087 keys)
hgu133aENTREZID has 20387 mapped keys (of 22283 keys)
```

```
> contents(hgu133aSYMBOL)[1:5]
$`1007_s_at`
[1] "DDR1"

$`1053_at`
[1] "RFC2"

$`117_at`
[1] "HSPA6"

$`121_at`
[1] "PAX8"

$`1255_g_at`
[1] "GUCA1A"

> probeset<-rownames(exprs(eset))
> probeset[1:5]
[1] "1007_s_at" "1053_at"   "117_at"    "121_at"    "1255_g_at"
>
```

RKWar
well as

Getti

Due to
introdu

The se
the abo

affylmGUI – Untitled

File RNA Targets Normalization Linear Model Evaluate TopTable Plot Help

Choose working directory

Choose working directory (containing input files):

/home/acb/old/biodata/cmap/tmp

Browse

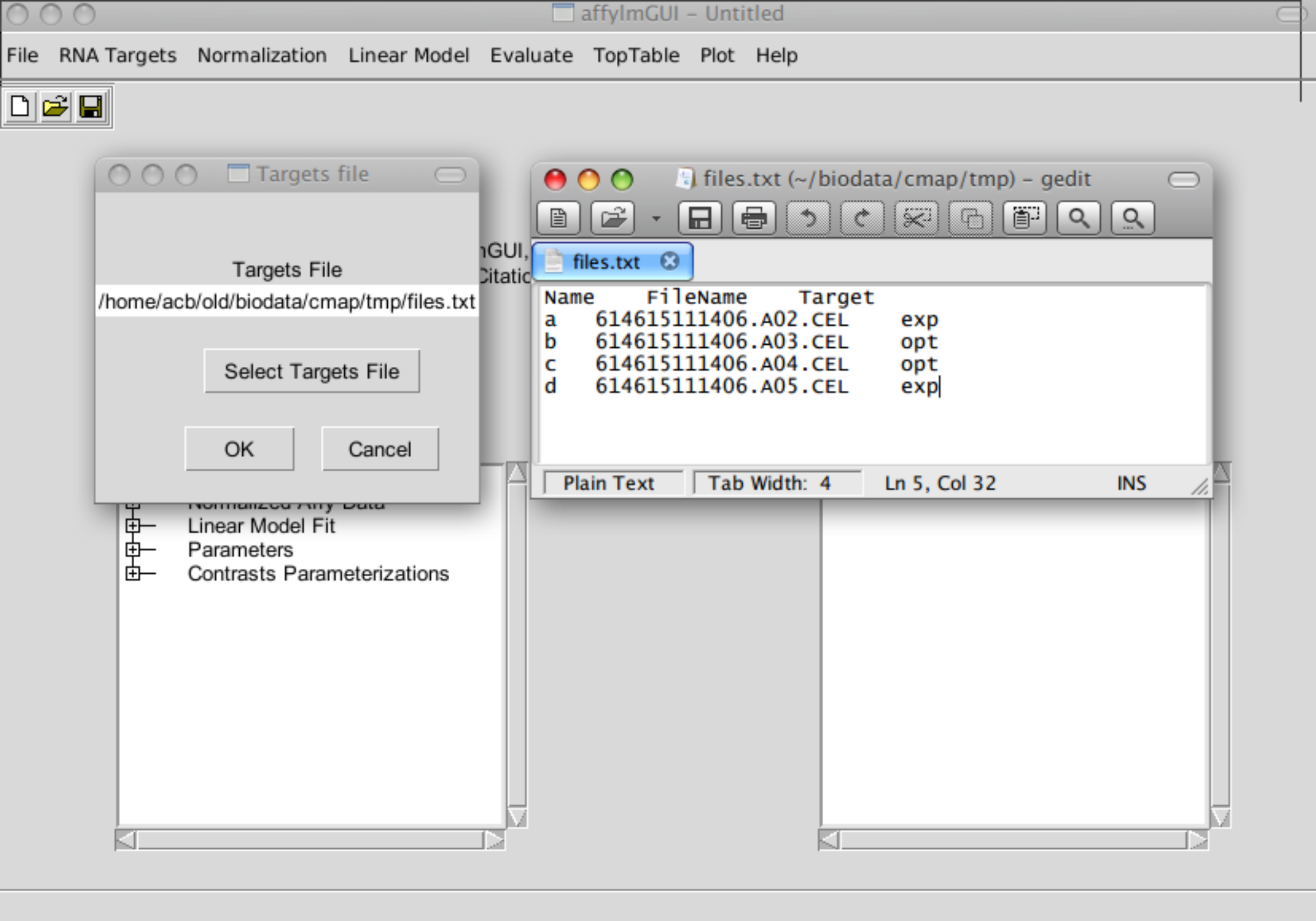
OK Cancel

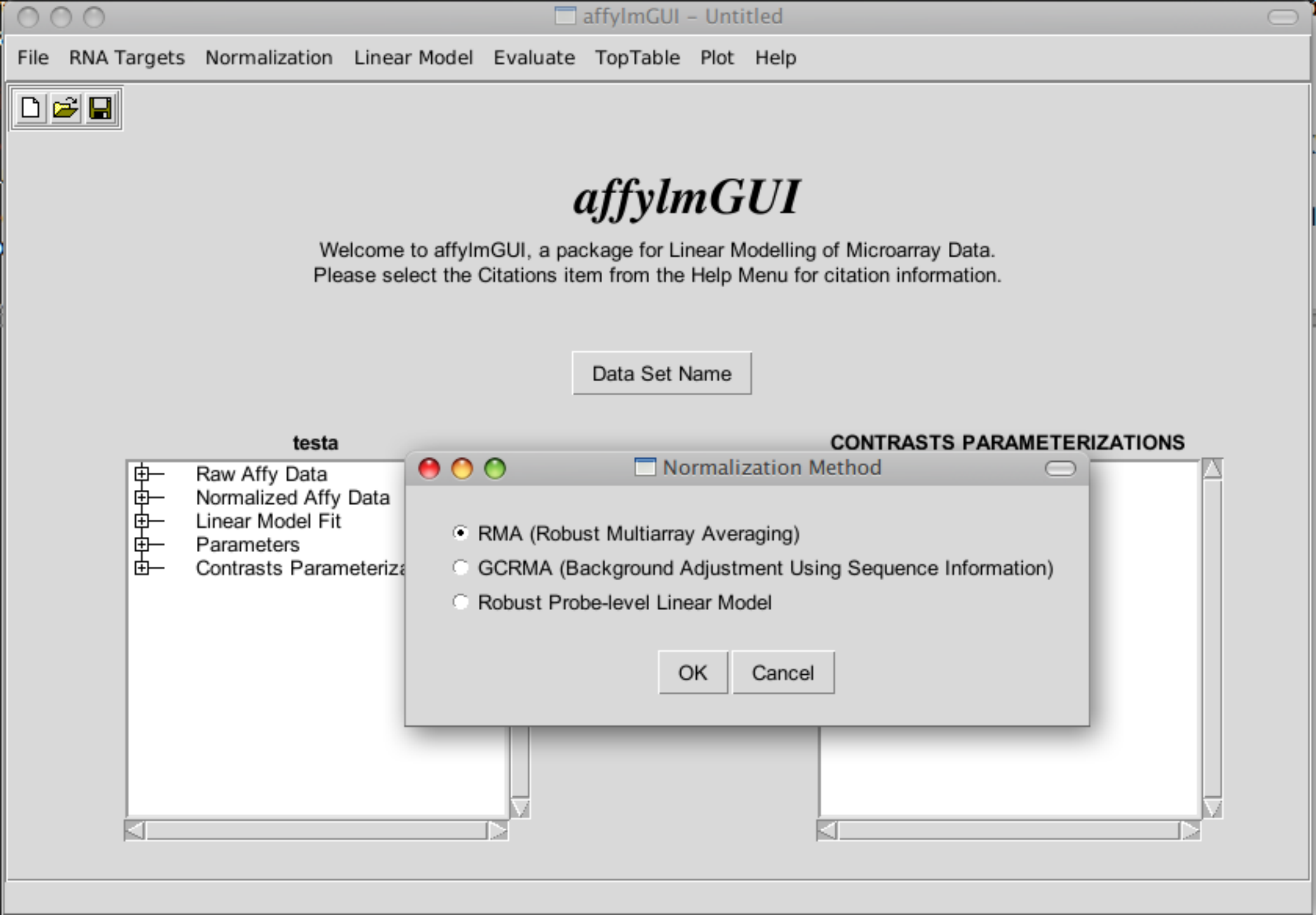
Untitled

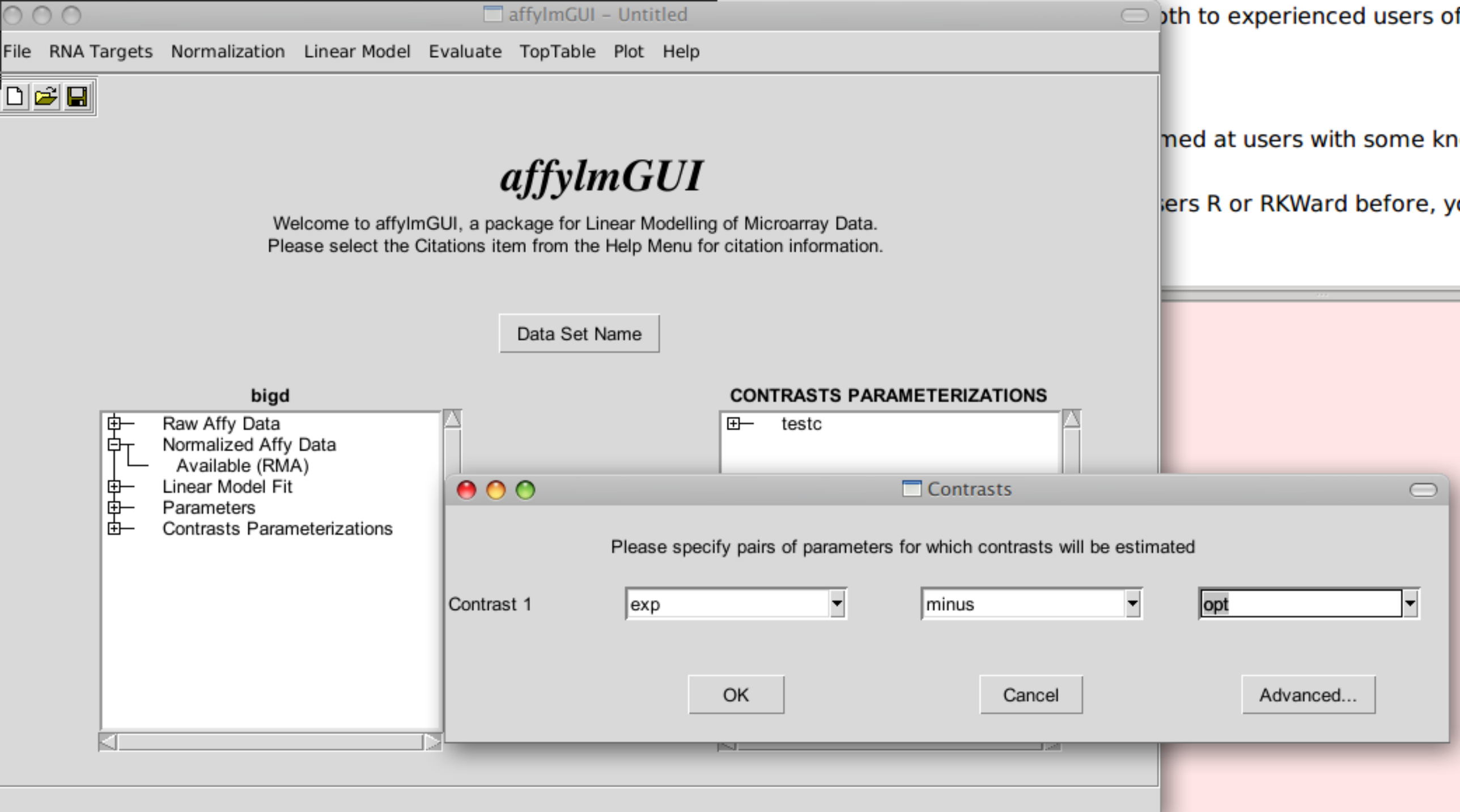
- Raw Affy Data
- Normalized Affy Data
- Linear Model Fit
- Parameters
- Contrasts Parameterizations

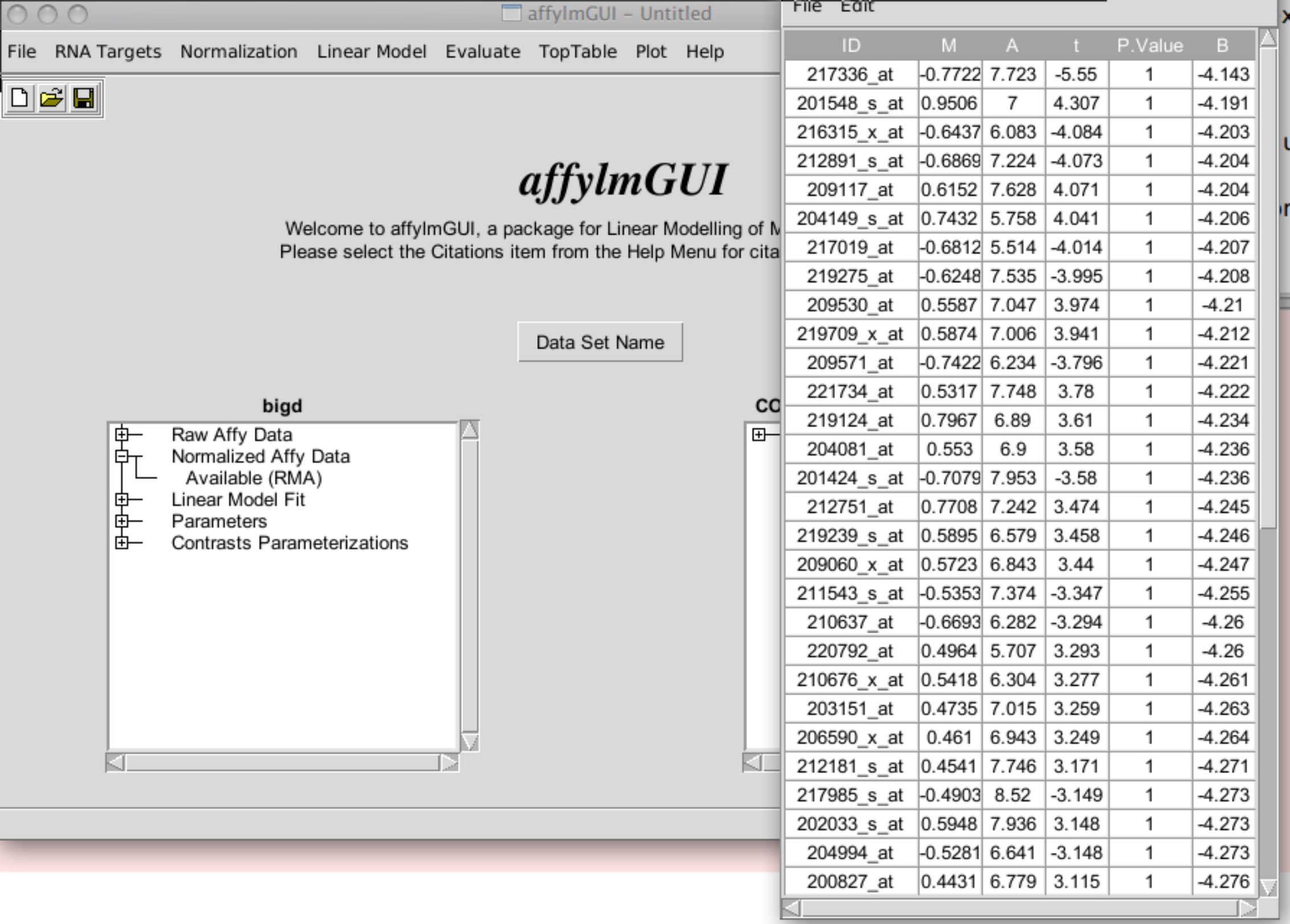
CONTRASTS PARAMETERIZATIONS

```
> library(affylmGUI)
> affylmGUI()

Searching for user-defined affylmGUI commands in /home/john/R/i486-pc-linux-gnu-library/2.15/affylmGUI/etc ...
>
```


affylmGUI - Untitled
File
RNA Targets
Normalization
Linear Model
Evaluate
TopTable
Plot
Help
Targets file
Targets File
/home/acb/old/biodata/cmap/tmp/files.txt
Select Targets File
OK
Cancel
files.txt (~/biodata/cmap/tmp) - gedit
files.txt
Name FileName Target
a 614615111406.A02.CEL exp
b 614615111406.A03.CEL opt
c 614615111406.A04.CEL opt
d 614615111406.A05.CEL exp
Plain Text
Tab Width: 4
Ln 5, Col 32
INS
Linear Model Fit
Parameters
Contrasts Parameterizations


affylmGUI – Untitled
File
RNA Targets
Normalization
Linear Model
Evaluate
TopTable
Plot
Help
affylmGUI
Welcome to affylmGUI, a package for Linear Modelling of Microarray Data.
Please select the Citations item from the Help Menu for citation information.
Data Set Name
testa
CONTRASTS PARAMETERIZATIONS
Raw Affy Data
Normalized Affy Data
Linear Model Fit
Parameters
Contrasts Parameteriza
Normalization Method
RMA (Robust Multiarray Averaging)
GCRMA (Background Adjustment Using Sequence Information)
Robust Probe-level Linear Model
OK
Cancel


affylmGUI – Untitled
File
RNA Targets
Normalization
Linear Model
Evaluate
TopTable
Plot
Help
affylmGUI
Welcome to affylmGUI, a package for Linear Modelling of Microarray Data.
Please select the Citations item from the Help Menu for citation information.
Data Set Name
bigd
Raw Affy Data
Normalized Affy Data
Available (RMA)
Linear Model Fit
Parameters
Contrasts Parameterizations
CONTRASTS PARAMETERIZATIONS
testc
Contrasts
Please specify pairs of parameters for which contrasts will be estimated
Contrast 1
exp
minus
opt
OK
Cancel
Advanced...

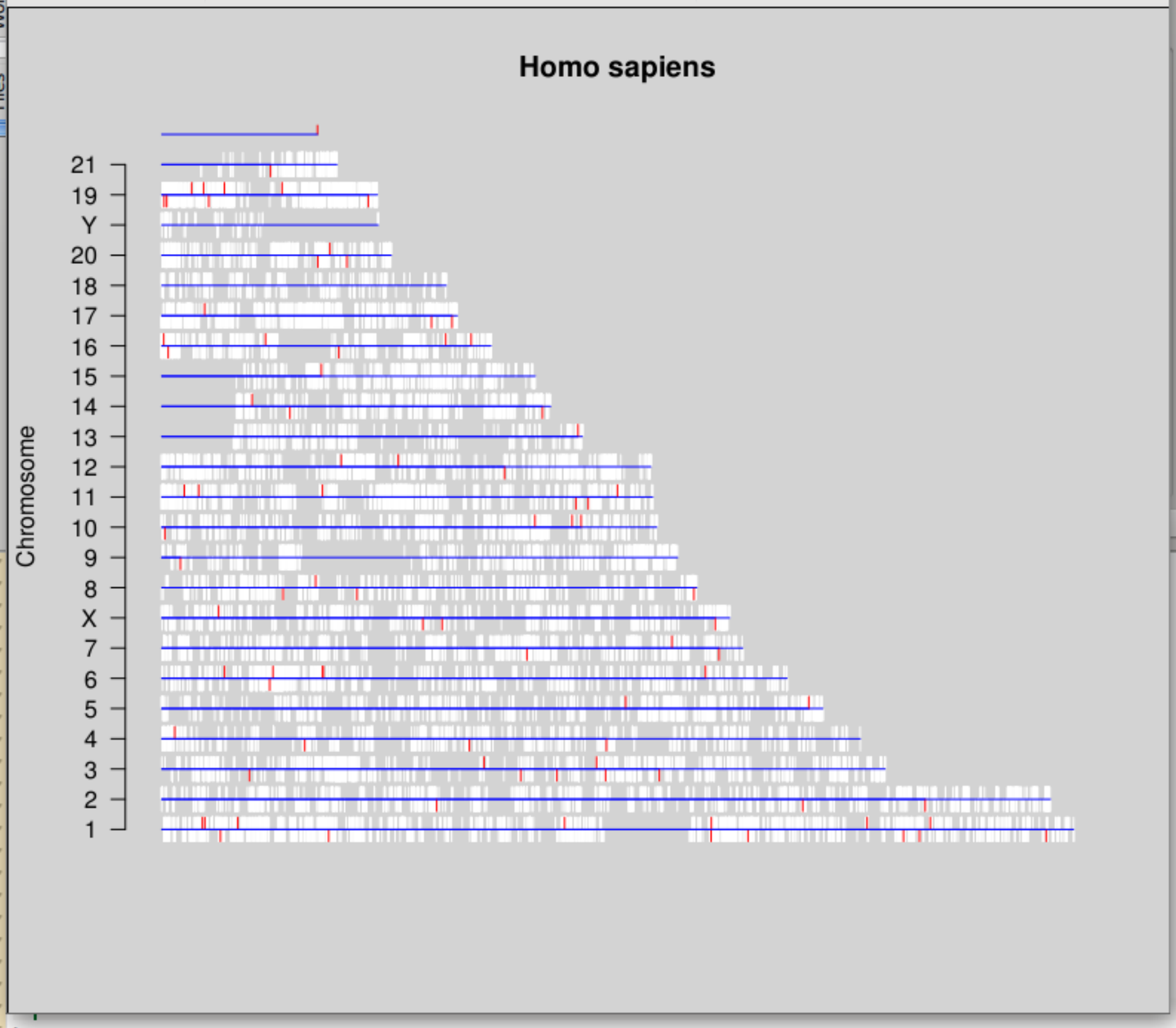


ID	M	A	t	P.Value	B
217336_at	-0.7722	7.723	-5.55	1	-4.143
201548_s_at	0.9506	7	4.307	1	-4.191
216315_x_at	-0.6437	6.083	-4.084	1	-4.203
212891_s_at	-0.6869	7.224	-4.073	1	-4.204
209117_at	0.6152	7.628	4.071	1	-4.204
204149_s_at	0.7432	5.758	4.041	1	-4.206
217019_at	-0.6812	5.514	-4.014	1	-4.207
219275_at	-0.6248	7.535	-3.995	1	-4.208
209530_at	0.5587	7.047	3.974	1	-4.21
219709_x_at	0.5874	7.006	3.941	1	-4.212
209571_at	-0.7422	6.234	-3.796	1	-4.221
221734_at	0.5317	7.748	3.78	1	-4.222
219124_at	0.7967	6.89	3.61	1	-4.234
204081_at	0.553	6.9	3.58	1	-4.236
201424_s_at	-0.7079	7.953	-3.58	1	-4.236
212751_at	0.7708	7.242	3.474	1	-4.245
219239_s_at	0.5895	6.579	3.458	1	-4.246
209060_x_at	0.5723	6.843	3.44	1	-4.247
211543_s_at	-0.5353	7.374	-3.347	1	-4.255
210637_at	-0.6693	6.282	-3.294	1	-4.26
220792_at	0.4964	5.707	3.293	1	-4.26
210676_x_at	0.5418	6.304	3.277	1	-4.261
203151_at	0.4735	7.015	3.259	1	-4.263
206590_x_at	0.461	6.943	3.249	1	-4.264
212181_s_at	0.4541	7.746	3.171	1	-4.271
217985_s_at	-0.4903	8.52	-3.149	1	-4.273
202033_s_at	0.5948	7.936	3.148	1	-4.273
204994_at	-0.5281	6.641	-3.148	1	-4.273
200827_at	0.4431	6.779	3.115	1	-4.276

R Graphics: Device 2 (ACTIVE) – RKWard

Device Edit View History Window Settings Help

Copy to output | Go to plot | Previous plot | Next plot | Remove this plot



rkward_welcome | toptablef | Ou

	1	2	3
Name	X	ID	M
Label			
Type	Number	Factor	Number
Format			
Levels		200054_at#,...	

1	16701	217336_at	-0.772243
2	1076	201548_s_at	0.950628
3	15687	216315_x_at	-0.643748
4	12276	212891_s_at	-0.686907
5	8611	209117_at	0.615163
6	3676	204149_s_at	0.743227

```
>
>
>
> library(annotate)
> library(geneplotter)
> library("hgu133a.db")
> newChrom <- buildChromLocation("hgu133a.db")
> cPlot(newChrom,c(1:21,'X','Y'))
> cColor(as.character(toptablef[["ID"]]),"red",newChrom)
```

```
> ?unlist
> library(GOstats); library(GO.db); library(ath1121501.db); library(annotate)
Loading required package: Biobase
Loading required package: BiocGenerics

Attaching package: 'BiocGenerics'

The following object(s) are masked from 'package:stats':

    xtabs

The following object(s) are masked from 'package:base':

    anyDuplicated, cbind, colnames, duplicated, eval, Filter, Find,
    get, intersect, lapply, Map, mapply, mget, order, paste, pmax,
    pmax.int, pmin, pmin.int, Position, rbind, Reduce, rep.int,
    rownames, sapply, setdiff, table, tapply, union, unique

Welcome to Bioconductor

    Vignettes contain introductory material; view with
    'browseVignettes()'. To cite Bioconductor, see
    'citation("Biobase")', and for packages 'citation("pkgname")'.

Loading required package: Category
Loading required package: AnnotationDbi
Loading required package: graph
Loading required package: DBI

Loading required package: org.At.tair.db


> GOTERM$"GO:0003700"
GOID: GO:0003700
Term: sequence-specific DNA binding transcription factor activity
Ontology: MF
Definition: Interacting selectively and non-covalently with a specific
    DNA sequence in order to modulate transcription. The transcription
    factor may or may not also interact selectively with a protein or
    macromolecular complex.
Synonym: transcription factor activity
Synonym: GO:0000130
Secondary: GO:0000130
> GOTERM$"GO:0003700"@Ontology
[1] "MF"
> GOTERM$"GO:0003700"@Term
[1] "sequence-specific DNA binding transcription factor activity"
> zz <- eapply(GOTERM, function(x) x@Ontology)
>
```

Upload List Background

Upload Gene List

Demolist 1 Demolist 2

Upload Help

Step 1: Enter Gene List

A: Paste a list

213809_x_at
203266_s_at
202912_at
215818_at

Clear

Or

B:Choose From a File

Choose File No file chosen

Multi-List File

Step 2: Select Identifier

AFFYMETRIX_3PRIME_IVT_ID

Step 3: List Type

Gene List
Background

Step 4: Submit List

Submit List

Analysis Wizard

Tell us how you like the tool
Contact us for questions

Step 1. Successfully submitted gene list

Current Gene List: List_1
Current Background: Human Genome U133A Array

Step 2. Analyze above gene list with one of DAVID tools

Which DAVID tools to use?

Functional Annotation Tool

- Functional Annotation Clustering
- Functional Annotation Chart
- Functional Annotation Table

Gene Functional Classification Tool

Gene ID Conversion Tool

Gene Name Batch Viewer

Upload List
Background

Population Manager

Select a background Help

HT Human Genome U133A
Homo sapiens
Human Genome U133A Arra

Select List to:

Use Rename

Affymetrix 3' IVT Backgrounds H

- Human Genome U133A 2 Array
- Human Genome U133A Array
- Human Genome U133B Array
- Human Genome U95A Array
- Human Genome U95Av2 Array

Affymetrix Exon Backgrounds He

- HuEx-1_0-st-v2
- HuGene-1_0-st-v1
- MoEx-1_0-st-v1
- MoGene-1_0-st-v1
- RaEx-1_0-st-v1

Affymetrix SNP Backgrounds Help

Open Create Save Cut Copy Paste Paste inside selection Paste

Workspace Files

All Non-Functions Functions

Show All Environments

Show Hidden Objects

Name	Label	Type	Class
+ package:annotate			
+ package:org.Hs.eg.db			
+ package:hgu133a.db			

Update

top

Name X
Label
1

```
> hgu133a.db
ChipDb object:
| DBSCHEMAVERSION: 2.1
| Db type: ChipDb
| Supporting package: AnnotationDbi
| DBSCHEMA: HUMANCHIP_DB
| ORGANISM: Homo sapiens
| SPECIES: Human
| MANUFACTURER: Affymetrix
| CHIPNAME: Human Genome U133 Set
| MANUFACTURERURL: http://www.affymetrix.com/support/technical/byproduct.affx?product=hgu133
| EGSOURCEDATE: 2012-Mar7
| EGSOURCENAME: Entrez Gene
| EGSOURCEURL: ftp://ftp.ncbi.nlm.nih.gov/gene/DATA
| CENTRALID: ENTREZID
| TAXID: 9606
| GOSOURCENAME: Gene Ontology
| GOSOURCEURL: ftp://ftp.geneontology.org/pub/go/godatabase/archive/latest-lite/
| GOSOURCEDATE: 20120303
```

Command log R Console Help search

Ready.

Step 1. Successfully submitted gene list

Current Gene List: List_1

Current Background: Human Genome U133A Array

Step 2. Analyze above gene list with one of DAVID tools

Which DAVID tools to use?

- Functional Annotation Tool
  - Functional Annotation Clustering
  - Functional Annotation Chart
  - Functional Annotation Table
- Gene Functional Classification Tool
- Gene ID Conversion Tool
- Gene Name Batch Viewer

Annotation Summary Results

Help and Tool Manual

Current Gene List: List_1 **99 DAVID IDs**

Current Background: Human Genome U133A Array **Check Defaults** ☑ Clear All

- ⊞ **Disease** (1 selected)
- ⊞ **Functional_Categories** (3 selected)
- ⊞ **Gene_Ontology** (3 selected)
- ⊞ **General Annotations** (0 selected)
- ⊞ **Literature** (0 selected)
- ⊞ **Main_Accessions** (0 selected)
- ⊞ **Pathways** (3 selected)
- ⊞ **Protein_Domains** (3 selected)
- ⊞ **Protein_Interactions** (0 selected)
- ⊞ **Tissue_Expression** (0 selected)

Red annotation categories denote DAVID defined defaults

Combined View for Selected Annotation

Functional Annotation Clustering

Functional Annotation Chart

Functional Annotation Table

Gene_Ontology (3 selected)

GOTERM_BP_1	74.7%	74	Chart	
GOTERM_BP_2	73.7%	73	Chart	
GOTERM_BP_3	70.7%	70	Chart	
GOTERM_BP_4	68.7%	68	Chart	
GOTERM_BP_5	64.6%	64	Chart	
GOTERM_BP_ALL	74.7%	74	Chart	
☑ GOTERM_BP_FAT	70.7%	70	Chart	
GOTERM_CC_1	79.8%	79	Chart	
GOTERM_CC_2	78.8%	78	Chart	
GOTERM_CC_3	78.8%	78	Chart	
GOTERM_CC_4	78.8%	78	Chart	
GOTERM_CC_5	78.8%	78	Chart	
GOTERM_CC_ALL	79.8%	79	Chart	
☑ GOTERM_CC_FAT	59.6%	59	Chart	
GOTERM_MF_1	75.8%	75	Chart	
GOTERM_MF_2	74.7%	74	Chart	
GOTERM_MF_3	63.6%	63	Chart	
GOTERM_MF_4	60.6%	60	Chart	
GOTERM_MF_5	49.5%	49	Chart	
GOTERM_MF_ALL	75.8%	75	Chart	
☑ GOTERM_MF_FAT	64.6%	64	Chart	
PANTHER_BP_ALL	58.6%	58	Chart	
PANTHER_MF_ALL	67.7%	67	Chart	

DAVID: Database for Annotation, Visualization, and Integrated Discovery (Laboratory of Immunopathogenesis and Bioinformatics (LIB); National Institute of Allergies and Infectious Dise

david.abcc.ncifcrf.gov/chartReport.jsp?annot=39

99 DAVID IDs

Options

Rerun Using Options | Create Sublist

6 chart records

Download

Sublist	Category	Term	RT	Genes	Count	%	P-Value	Benjam
	GOTERM_MF_FAT	calmodulin binding	RT		5	5.1	1.1E-2	8.7E-1
	GOTERM_MF_FAT	specific transcriptional repressor activity	RT		3	3.0	1.7E-2	8.0E-1
	GOTERM_MF_FAT	transcription repressor activity	RT		6	6.1	3.5E-2	9.0E-1
	GOTERM_MF_FAT	DNA-dependent ATPase activity	RT		3	3.0	4.8E-2	9.1E-1
	GOTERM_MF_FAT	DNA binding	RT		17	17.2	7.0E-2	9.4E-1
	GOTERM_MF_FAT	magnesium ion binding	RT		6	6.1	8.0E-2	9.3E-1

72 gene(s) from your list are not in the output.

General Annotations (0 selected)

Literature (0 selected)

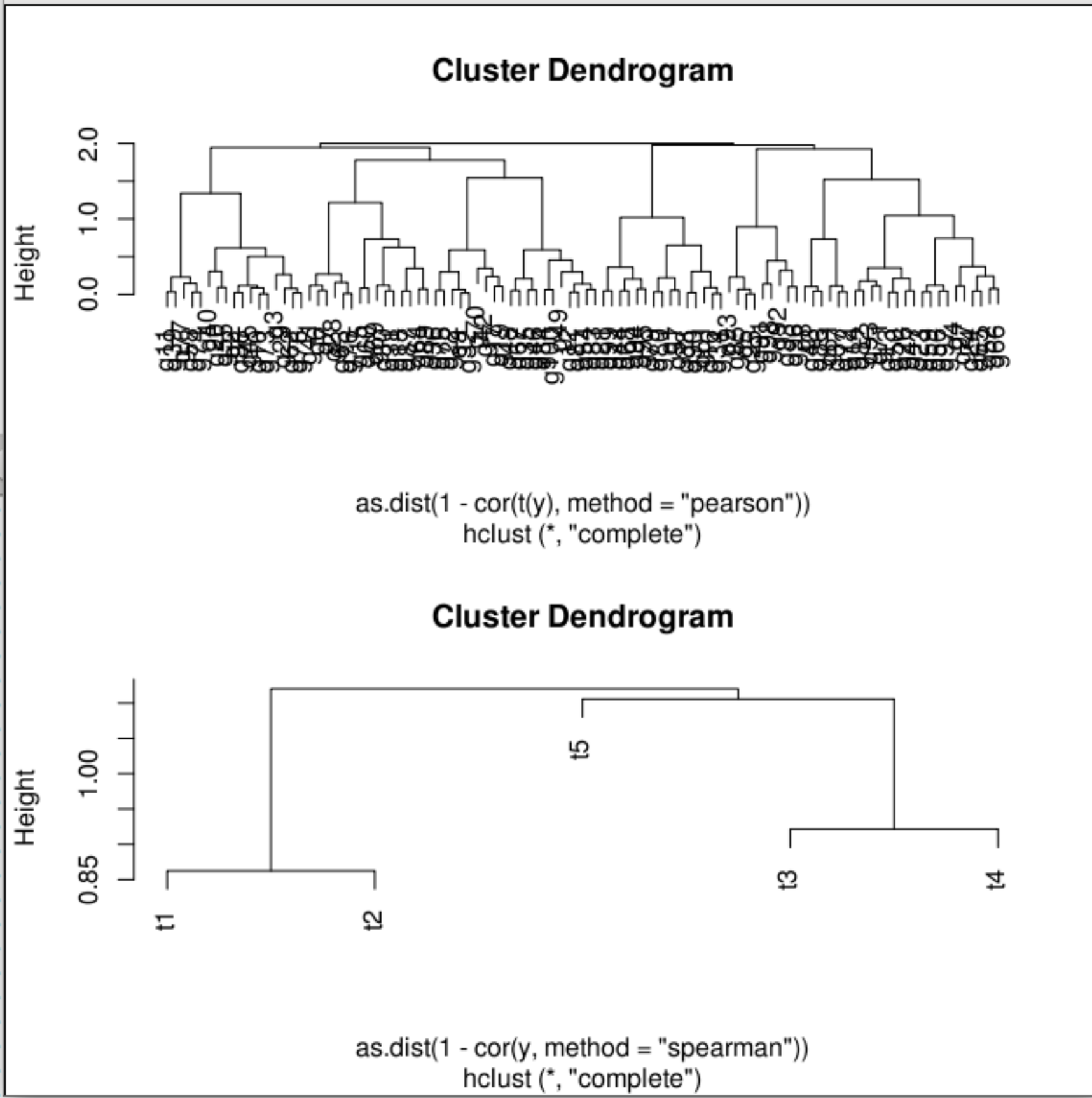
	t1	t2	t3	t4	t5
g1	-1.04520178	-0.380935801	0.32253285	0.188196643	0.28007615
g2	0.30260081	-0.970551319	2.59950534	1.176315379	0.18996084
g3	0.15975841	-0.478759036	1.73153639	1.061661475	1.57966492
g4	-0.78985782	0.081254590	-0.53945136	0.234886833	0.77129062
g5	-0.07311381	-0.375386205	-0.40928292	-1.308708764	-0.63216358
g6	-2.34589634	0.840200643	0.66167230	-1.007925101	-1.21183069
g7	-0.87402211	0.733826049	1.25435683	-0.294792532	-0.63150717
g8	1.37970856	-0.586133357	-0.92211485	-0.652446545	-0.04455500
g9	-0.38065076	0.303324485	-1.08968012	0.418261480	-1.33007870
g10	-0.18620830	-0.841703730	0.03012982	-0.278655199	-0.64582184
g11	1.14214019	0.053108651	0.23298126	0.899382537	-0.80270697
g12	1.44513788	-0.558154629	2.04270902	0.321967760	-1.54073249
g13	0.99803579	-0.776420713	-0.43827247	1.243851533	1.35663368
g14	0.96289504	-0.007877564	-1.09371899	1.340719288	1.60252259
g15	-1.32178711	1.120730238	0.37916397	1.248446497	-0.46664231
g16	-0.56333368	-1.093202573	-0.24292916	1.186015185	1.16267207
g17	-0.44602650	1.497993750	-1.32988839	0.180666341	-0.66531597
g18	-0.59273243	0.809950819	0.91060528	0.828761280	-1.71046485
g19	0.83650966	2.229876451	-0.81521291	0.081180504	0.20237456
g20	1.20476925	-1.801373447	0.48730660	1.276596164	0.25094667
g21	-1.91517457	-0.611499880	-0.12086295	-1.251266330	0.88628046
g22	0.18428016	1.182773003	0.39024382	0.638688662	1.05100182

```
> hr <- hclust(as.dist(1-cor(t(y), method="pearson")), method="complete")
> hr

Call:
hclust(d = as.dist(1 - cor(t(y), method = "pearson")), method = "complete")

Cluster method   : complete
Number of objects: 100

> hc <- hclust(as.dist(1-cor(y, method="spearman")), method="complete")
> hc

Call:
hclust(d = as.dist(1 - cor(y, method = "spearman")), method = "complete")

Cluster method   : complete
Number of objects: 5

> str(hc)
List of 7
 $ merge      : int [1:4, 1:2] -1 -3 -5 1 -2 -4 2 3
 $ height     : num [1:4] 0.862 0.921 1.106 1.12
 $ order      : int [1:5] 1 2 5 3 4
 $ labels     : chr [1:5] "t1" "t2" "t3" "t4" ...
 $ method     : chr "complete"
 $ call       : language hclust(d = as.dist(1 - cor(y, method = "spearman")), method = "complete")
 $ dist.method: NULL
 - attr(*, "class")= chr "hclust"
>
```

R Graphics: Device 2 (ACTIVE) - RKWard

Device Edit View History Window Settings Help

Copy to output | Go to plot | Previous plot | Next plot | Remove this plot

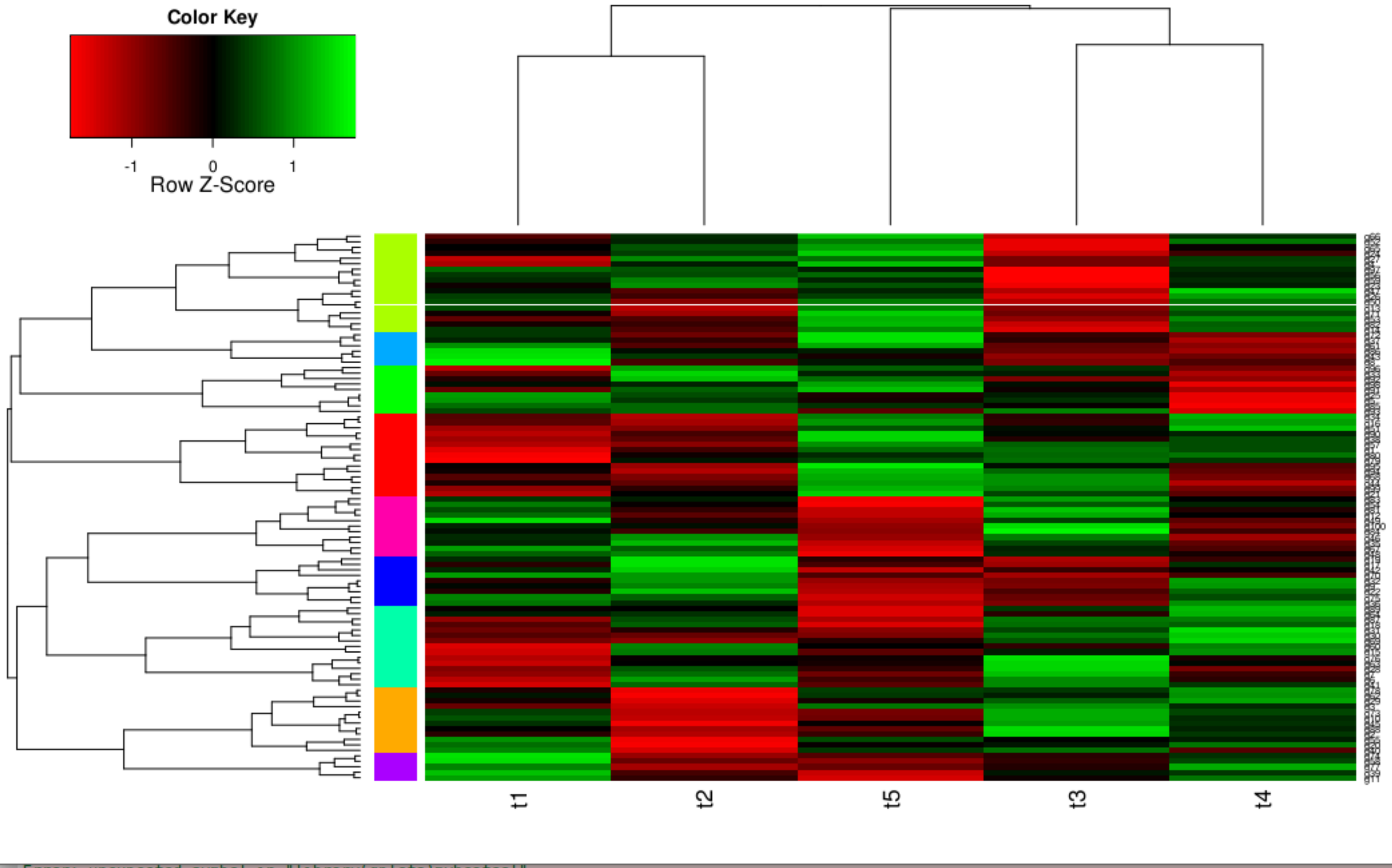



```
par(mfrow = c(2, 1))
plot(hr)
plot(hc)
```

```
> mycl <- cutree(hr, h=max(hr$height)/1.5)
> mycl
  g1   g2   g3   g4   g5   g6   g7   g8   g9  g10  g11  g12  g13  g14  g15  g16
   1    2    2    3    4    5    5    6    7    2    8    9    3    3    5    1
 g17  g18  g19  g20  g21  g22  g23  g24  g25  g26  g27  g28  g29  g30  g31  g32
   7    5    7    2    1    7    3    3    4    3    3    5    2    5    5    7
 g33  g34  g35  g36  g37  g38  g39  g40  g41  g42  g43  g44  g45  g46  g47  g48
   4    1    9    7    6    1    8    2    5    7    6    1    2    9    3    9
 g49  g50  g51  g52  g53  g54  g55  g56  g57  g58  g59  g60  g61  g62  g63  g64
   9    3    1    3    3    9    2    3    1    8    3    5    6    2    5    5
 g65  g66  g67  g68  g69  g70  g71  g72  g73  g74  g75  g76  g77  g78  g79  g80
   3    3    9    1    5    7    3    6    2    8    7    5    8    2    1    1
 g81  g82  g83  g84  g85  g86  g87  g88  g89  g90  g91  g92  g93  g94  g95  g96
   9    3    9    9    4    6    5    2    5    1    4    4    4    1    1    4
 g97  g98  g99 g100
   3    4    1    9
> ?rainbow
> mycolhc <- rainbow(length(unique(mycl)))
> mycolhc <- mycolhc[as.vector(mycl)]
> mycolhc
  [1] "#FF0000FF" "#FFAA00FF" "#FFAA00FF" "#AAFF00FF" "#00FF00FF" "#00FFAAFF"
  [7] "#00FFAAFF" "#00AAFFFF" "#0000FFFF" "#FFAA00FF" "#AA00FFFF" "#FF00AAFF"
 [13] "#AAFF00FF" "#AAFF00FF" "#00FFAAFF" "#FF0000FF" "#0000FFFF" "#00FFAAFF"
 [19] "#0000FFFF" "#FFAA00FF" "#FF0000FF" "#0000FFFF" "#AAFF00FF" "#AAFF00FF"
 [25] "#00FF00FF" "#AAFF00FF" "#AAFF00FF" "#00FFAAFF" "#FFAA00FF" "#00FFAAFF"
 [31] "#00FFAAFF" "#0000FFFF" "#00FF00FF" "#FF0000FF" "#FF00AAFF" "#0000FFFF"
 [37] "#00AAFFFF" "#FF0000FF" "#AA00FFFF" "#FFAA00FF" "#00FFAAFF" "#0000FFFF"
 [43] "#00AAFFFF" "#FF0000FF" "#FFAA00FF" "#FF00AAFF" "#AAFF00FF" "#FF00AAFF"
 [49] "#FF00AAFF" "#AAFF00FF" "#FF0000FF" "#AAFF00FF" "#AAFF00FF" "#FF00AAFF"
 [55] "#FFAA00FF" "#AAFF00FF" "#FF0000FF" "#AA00FFFF" "#AAFF00FF" "#00FFAAFF"
 [61] "#00AAFFFF" "#FFAA00FF" "#00FFAAFF" "#00FFAAFF" "#AAFF00FF" "#AAFF00FF"
 [67] "#FF00AAFF" "#FF0000FF" "#00FFAAFF" "#0000FFFF" "#AAFF00FF" "#00AAFFFF"
 [73] "#FFAA00FF" "#AA00FFFF" "#0000FFFF" "#00FFAAFF" "#AA00FFFF" "#FFAA00FF"
 [79] "#FF0000FF" "#FF0000FF" "#FF00AAFF" "#AAFF00FF" "#FF00AAFF" "#FF00AAFF"
 [85] "#00FF00FF" "#00AAFFFF" "#00FFAAFF" "#FFAA00FF" "#00FFAAFF" "#FF0000FF"
 [91] "#00FF00FF" "#00FF00FF" "#00FF00FF" "#FF0000FF" "#FF0000FF" "#00FF00FF"
 [97] "#AAFF00FF" "#00FF00FF" "#FF0000FF" "#FF00AAFF"
>
```

Device Edit View History Window Settings Help

Copy to output | Go to plot | Previous plot | Next plot | Remove this plot

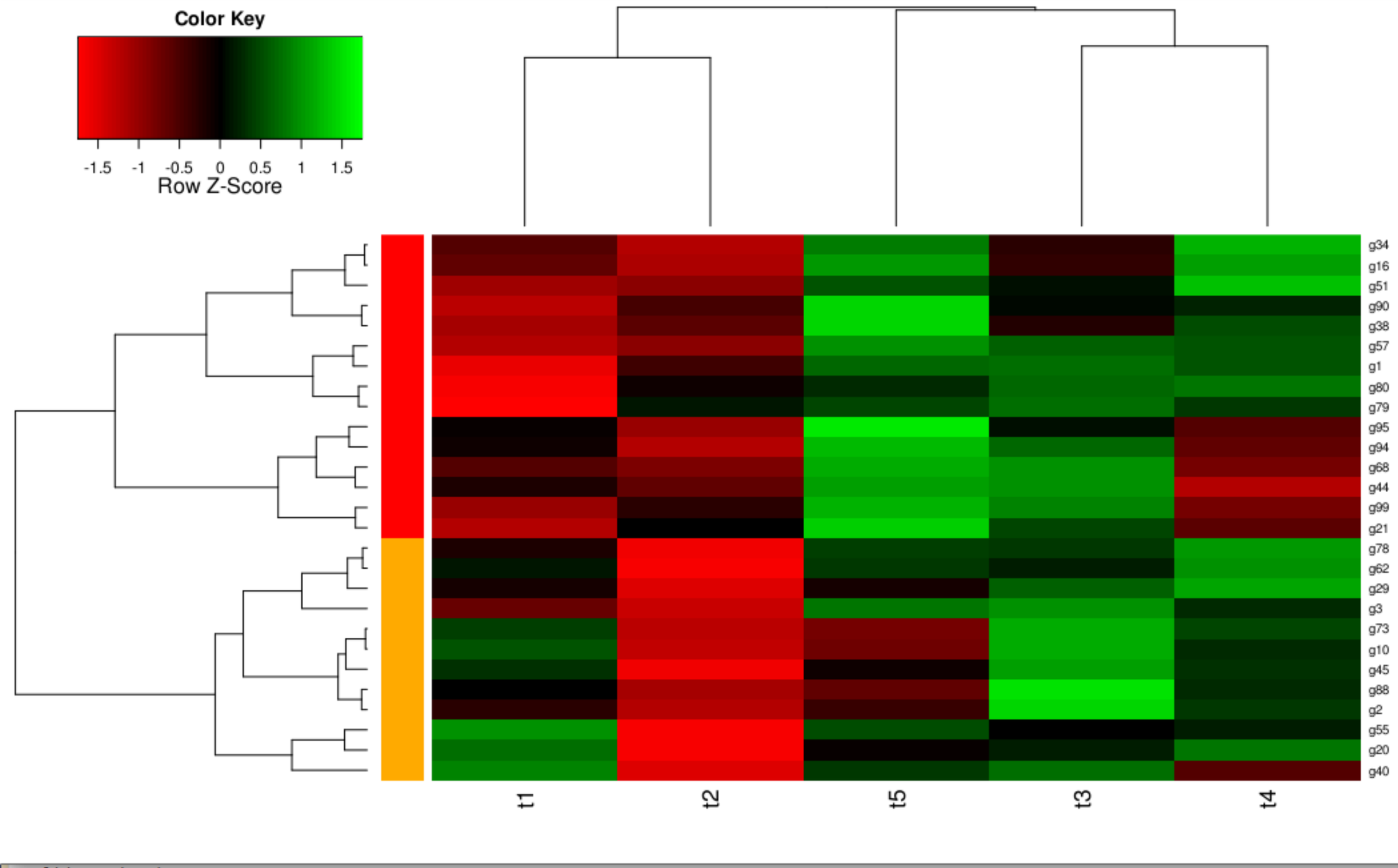



```
Error: unexpected symbol in "library(gplots)myheatcol"
> library(gplots)
> myheatcol <- redgreen(75)
> myheatcol
 [1] "#FF0000" "#F80000" "#F10000" "#EA0000" "#E30000" "#DD0000" "#D60000"
 [8] "#CF0000" "#C80000" "#C10000" "#BA0000" "#B30000" "#AC0000" "#A50000"
[15] "#9F0000" "#980000" "#910000" "#8A0000" "#830000" "#7C0000" "#750000"
[22] "#6E0000" "#670000" "#600000" "#5A0000" "#530000" "#4C0000" "#450000"
[29] "#3E0000" "#370000" "#300000" "#290000" "#220000" "#1C0000" "#150000"
[36] "#0E0000" "#070000" "#000000" "#000700" "#000E00" "#001500" "#001C00"
[43] "#002200" "#002900" "#003000" "#003700" "#003E00" "#004500" "#004C00"
[50] "#005300" "#005A00" "#006000" "#006700" "#006E00" "#007500" "#007C00"
[57] "#008300" "#008A00" "#009100" "#009800" "#009F00" "#00A500" "#00AC00"
[64] "#00B300" "#00BA00" "#00C100" "#00C800" "#00CF00" "#00D600" "#00DD00"
[71] "#00E300" "#00EA00" "#00F100" "#00F800" "#00FF00"
> heatmap.2(y, Rowv=as.dendrogram(hr), Colv=as.dendrogram(hc), col=myheatcol, scale="row", density.info="none", trace="none", RowSideColors=mycolhc) # C
>
```

R Graphics: Device 2 (ACTIVE) – RKWard

Device Edit View History Window Settings Help

Copy to output | Go to plot | Previous plot | Next plot | Remove this plot



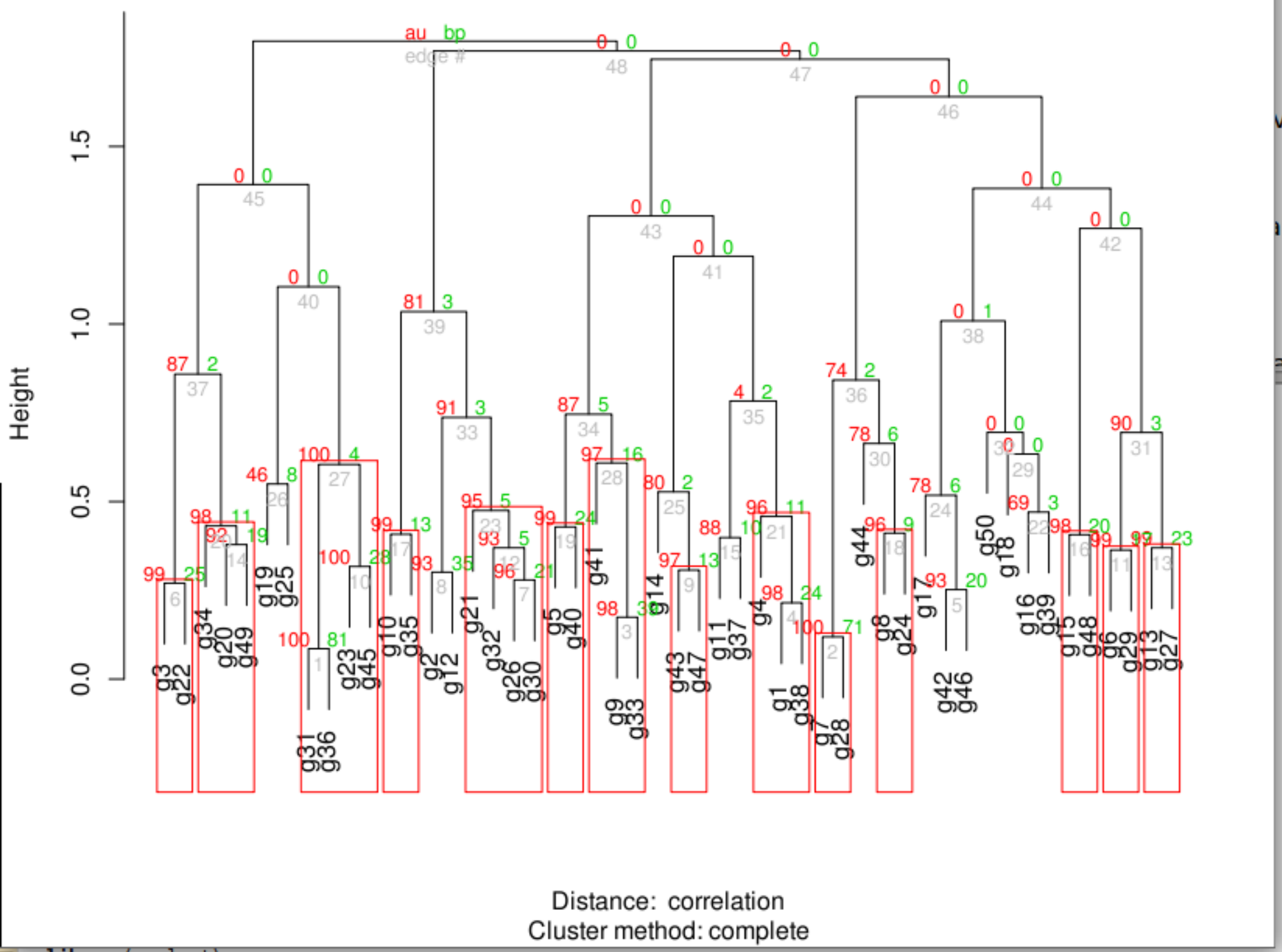
Object Viewer: y

colors.

start = 0, e

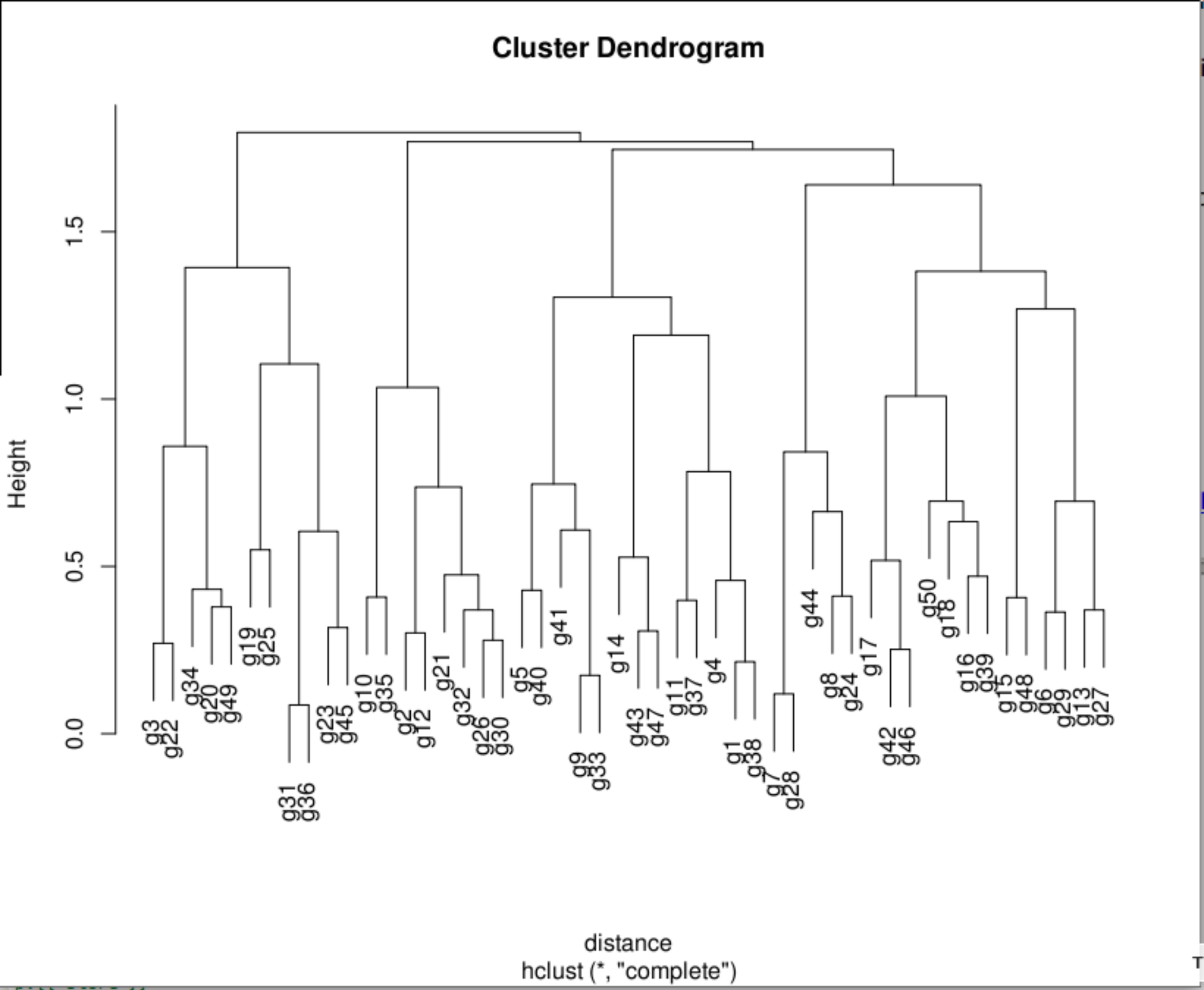
1)

```
> clid <- c(1,2)
> ysub <- y[names(mycl[mycl%in%clid]),]
> mycl[mycl%in%clid]
 g1  g2  g3 g10 g16 g20 g21 g29 g34 g38 g40 g44 g45 g51 g55 g57 g62 g68 g73 g78
  1   2   2   2   1   2   1   2   1   1   2   1   2   1   2   1   2   1   2   2
g79 g80 g88 g90 g94 g95 g99
  1   1   2   1   1   1   1
> names(mycl[mycl%in%clid])
 [1] "g1"  "g2"  "g3"  "g10" "g16" "g20" "g21" "g29" "g34" "g38" "g40" "g44"
[13] "g45" "g51" "g55" "g57" "g62" "g68" "g73" "g78" "g79" "g80" "g88" "g90"
[25] "g94" "g95" "g99"
> hrsub <- hclust(as.dist(1-cor(t(ysub), method="pearson")), method="complete")
> heatmap.2(ysub, Rowv=as.dendrogram(hrsub), Colv=as.dendrogram(hc), col=myheatcol, scale="row", density.info="none", trace="none", RowSideColors=mycolhc[mycl%in%clid])
>
```

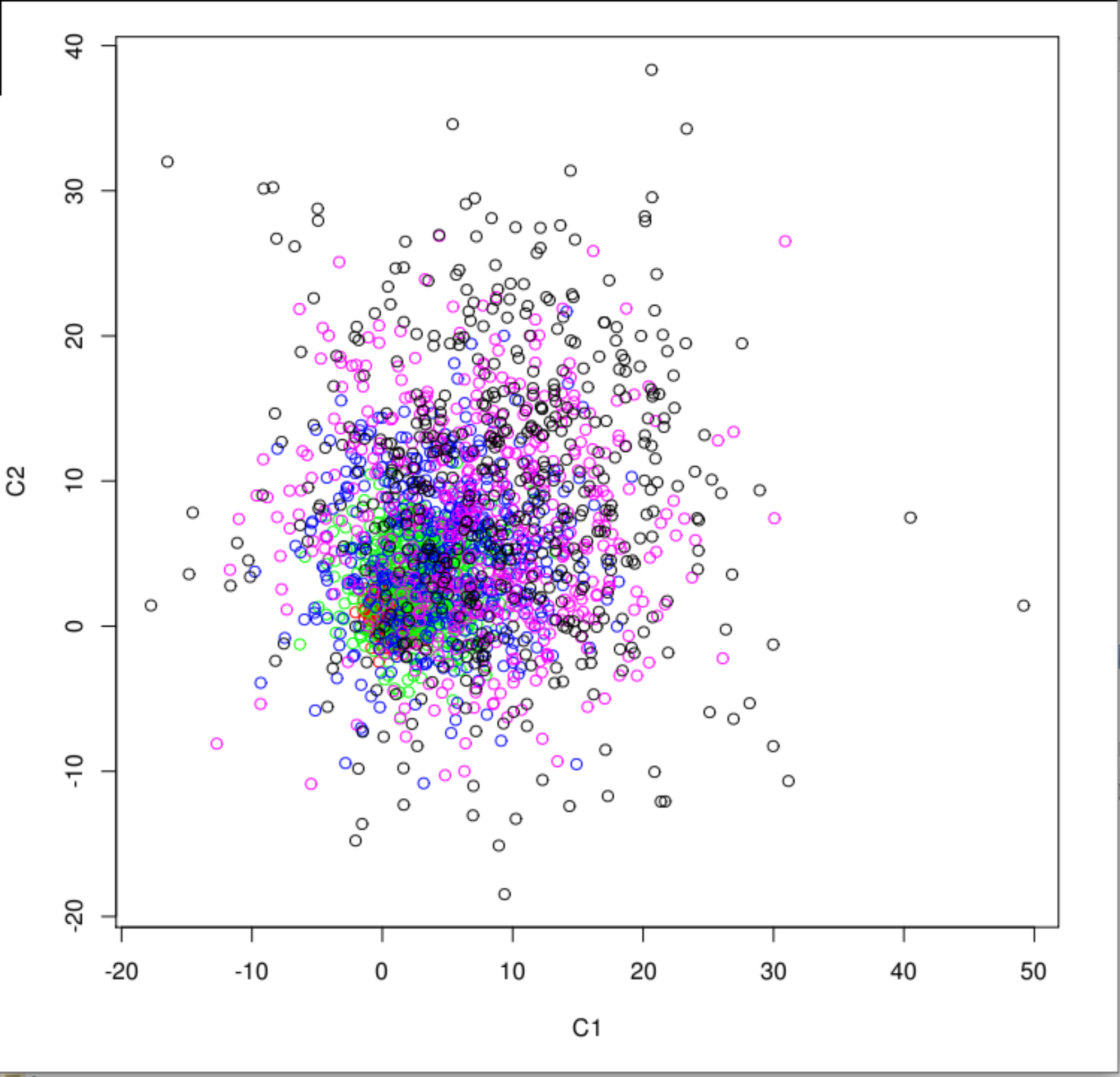



```
> library(pvclust)
> pv <- pvclust(scale(t(y)), method.dist="correlation", method.hclust="complete", nboot=100)
Bootstrap (r = 0.5)... Done.
Bootstrap (r = 0.6)... Done.
Bootstrap (r = 0.7)... Done.
Bootstrap (r = 0.8)... Done.
Bootstrap (r = 0.9)... Done.
Bootstrap (r = 1.0)... Done.
Bootstrap (r = 1.1)... Done.
Bootstrap (r = 1.2)... Done.
Bootstrap (r = 1.3)... Done.
Bootstrap (r = 1.4)... Done.
> plot(pv)
> pvrect(pv, alpha=0.95)
>
```

```
> pvpick(pv, alpha=0.95, pv="au", type="geq", max.only=TRUE)
$clusters
$clusters[[1]]
[1] "g7"  "g28"

$clusters[[2]]
[1] "g3"  "g22"

$clusters[[3]]
[1] "g43" "g47"

$clusters[[4]]
[1] "g6"  "g29"

$clusters[[5]]
[1] "g13" "g27"

$clusters[[6]]
[1] "g15" "g48"
```




```
> plot(pv[["hclust"]])
```

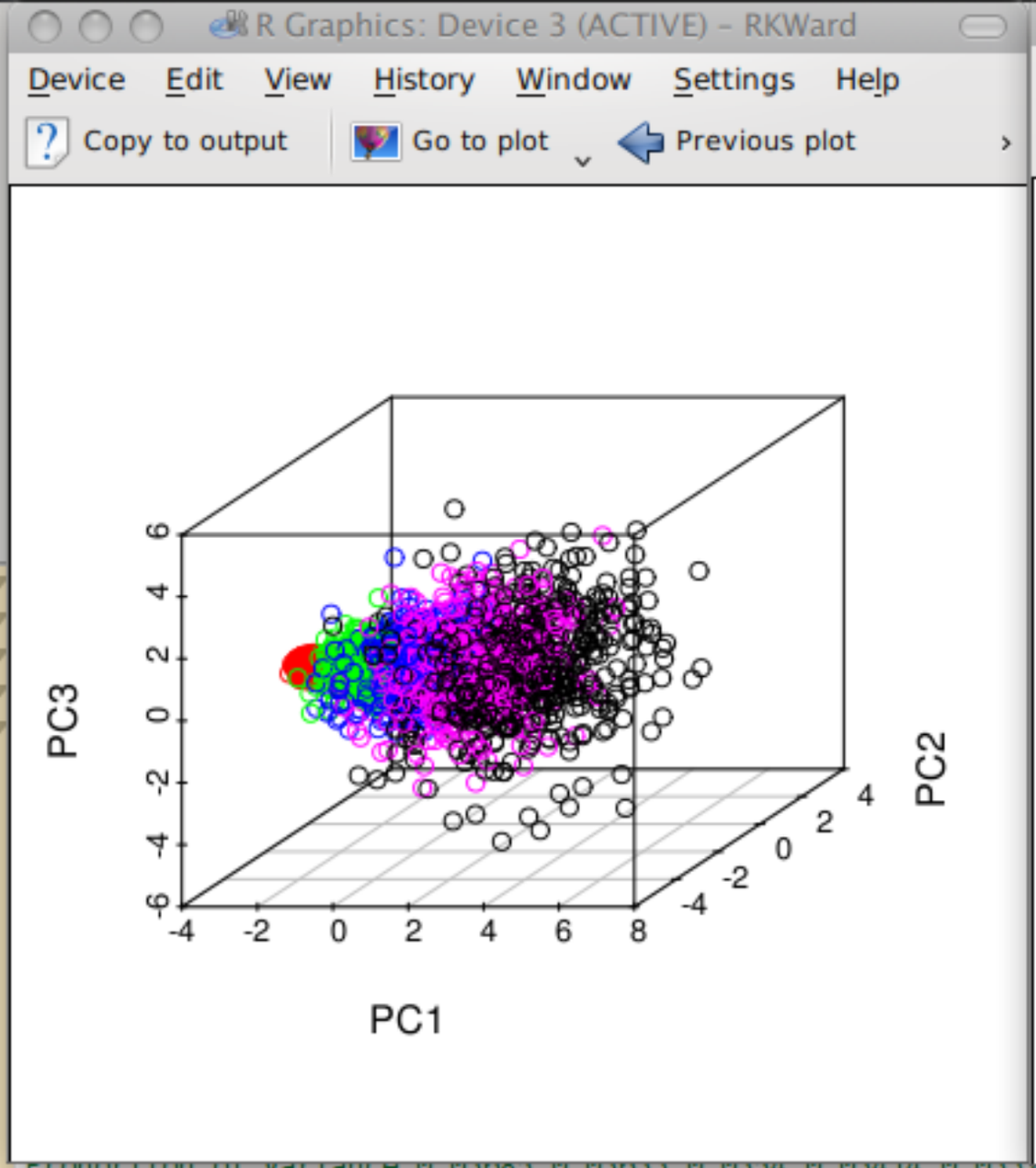


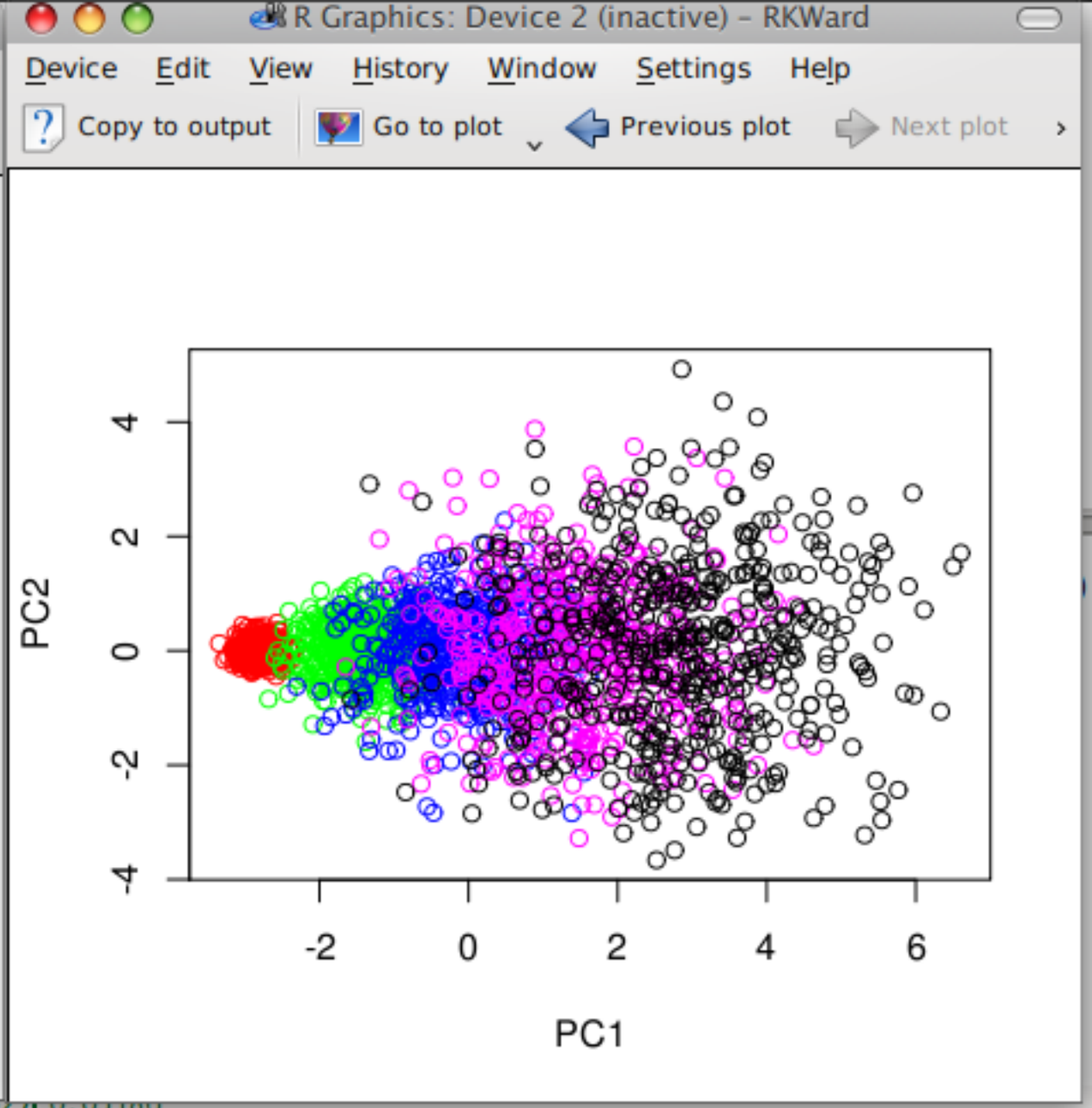
summary (x)

Ready

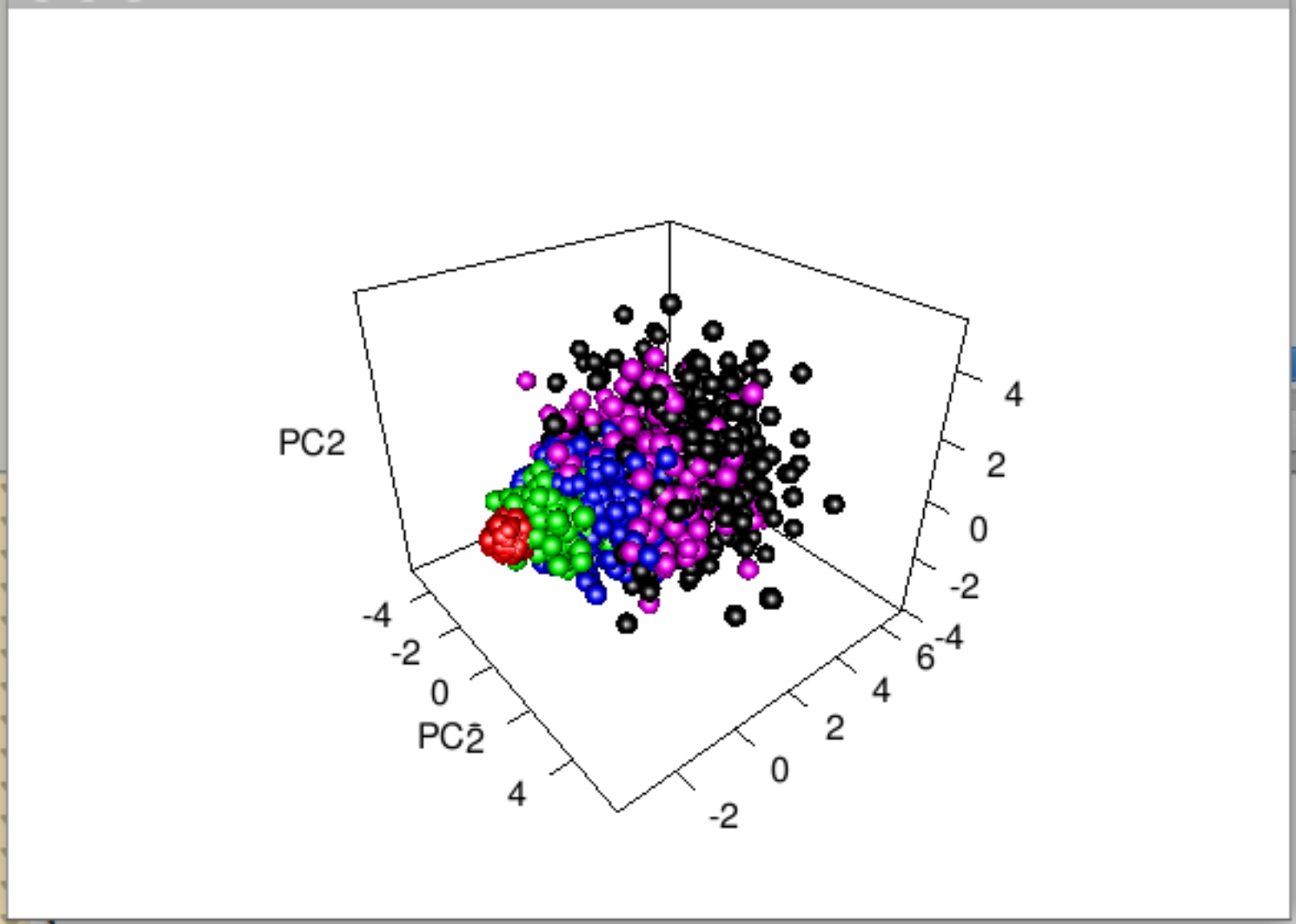
```
       C1
Min.   :-16.
1st Qu.:  0.
Median :  3.
Mean   :  4.
3rd Qu.:  8.
Max.   : 36.
       C5
Min.   :-20.
1st Qu.:  0.
Median :  3.
Mean   :  5.
3rd Qu.:  8.
Max.   : 38.
       C9
Min.   :-19.
1st Qu.:  0.
Median :  3.
Mean   :  4.
3rd Qu.:  7.
Max.   : 39.
      C13
Min.   :-19.
1st Qu.:  0.
Median :  3.
Mean   :  5.
3rd Qu.:  8.
Max.   : 31.
      C17
Min.   :-17.
1st Qu.:  0.
```

```
>
> z1 <- rnorm(10000, mean=1, sd=1)
> z2 <- rnorm(10000, mean=3, sd=3)
> z3 <- rnorm(10000, mean=5, sd=5)
> z4 <- rnorm(10000, mean=7, sd=7)
> z5 <- rnorm(10000, mean=9, sd=9)
> mydata <- matrix(c(z1, z2, z3, z4, z5), 2500, 20, byrow=T, dimnames=list(paste("R", 1:2500, sep=""), paste("C", 1:20, sep="")))
> plot(mydata,col=mycolors[sort(rep(1:5, 500))])
>
```






```
Proportion of Variance 0.03683 0.03633 0.0354 0.03414 0.03327 0.03186
Cumulative Proportion  0.82900 0.86533 0.9007 0.93487 0.96814 1.00000
> plot(pca$x,col=mycolors[sort(rep(1:5, 500))])
> x11()
> library(scatterplot3d)
> scatterplot3d(pca$x[,1:3],color=mycolors[sort(rep(1:5, 500))])
>
```




```
> library(scatterplot3d)
> rgl.open(); offset <- 50; par3d(windowRect=c(offset, offset, 640+offset, 640+offset))
> rgl.open();
> spheres3d(pca$x[,1], pca$x[,2], pca$x[,3], radius=0.3, color=mycolors[sort(rep(1:5, 500))], alpha=1, shininess=20)
> axes3d(col='black')
> title3d("", "", "PC1", "PC2", "PC3", col='black')
> bg3d("white")
> rgl.viewpoint(theta=45, phi=30, fov=60, zoom=1)
```

R: KEGG

KEGG PATHWAY Database

Wiring diagrams of molecular interactions, reactions, and relations

Pathway Maps

KEGG PATHWAY is a collection of manually drawn pathway maps (see new maps and update history) representing our knowledge on the molecular interaction and reaction networks for:

0. Global Map
1. Metabolism
Carbohydrate Energy Lipid Nucleotide Amino acid Other amino acid Glycan
Cofactor/vitamin Terpenoid/PK Other secondary metabolite Xenobiotics Overview
2. Genetic Information Processing
3. Environmental Information Processing
4. Cellular Processes
5. Organismal Systems
6. Human Diseases

and also on the structure relationships (KEGG drug structure maps) in:

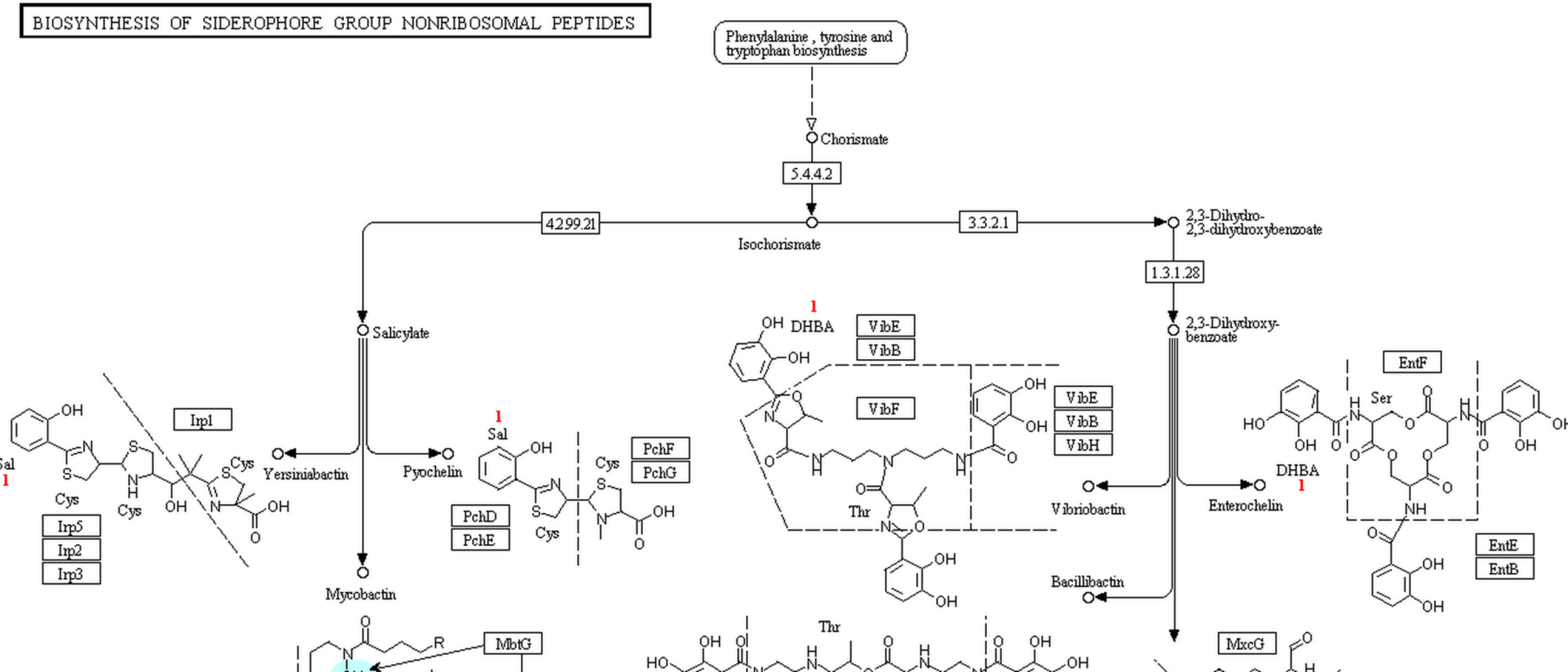
7. Drug Development

Pathway Mapping

KEGG PATHWAY mapping is the process to map molecular datasets, especially large-scale datasets in genomics, transcriptomics, proteomics, and metabolomics, to the KEGG pathway maps for biological interpretaion of higher-level systemic functions.

- Search Pathway - basic pathway mapping tool
- Search&Color Pathway - advanced pathway mapping tool
- Color Pathway - selected pathway map coloring tool

0. Global Map


BIOSYNTHESIS OF SIDEROPHORE GROUP NONRIBOSOMAL PEPTIDES
Phenylalanine , tyrosine and tryptophan biosynthesis
Chorismate
5.4.4.2
4.2.99.21
Isochorismate
3.3.2.1
2,3-Dihydro-2,3-dihydroxybenzoate
1.3.1.28
2,3-Dihydroxy-benzoate
Salicylate
DHBA
VibE
VibB
VibF
VibH
Irp1
Sal
Cys
Irp5
Irp2
Irp3
Yersiniabactin
Pyochelin
PchF
PchG
PchD
PchE
Mycobactin
Thr
Vibriobactin
Enterochelin
EntF
Ser
EntE
EntB
Bacillibactin
MbtG
MxcG
R

```
> library(graphite)
> kegg[[1]]
"ABC transporters" pathway from KEGG
Number of nodes     = 44
Number of edges     = 0
Type of identifiers = native
Retrieved on        = 2011-05-05
> kegg[[2]]
"Acute myeloid leukemia" pathway from KEGG
Number of nodes     = 58
Number of edges     = 167
Type of identifiers = native
Retrieved on        = 2011-05-05
> length(kegg)
[1] 232
> biocarta[[1]]
"acetylation and deacetylation of rela in nucleus" pathway from BioCarta
Number of nodes     = 6
Number of edges     = 9
Type of identifiers = native
Retrieved on        = 2011-05-12
> length(biocarta)
[1] 254
> reactome[[1]]
"2-LTR circle formation" pathway from Reactome
Number of nodes     = 142
Number of edges     = 8485
Type of identifiers = native
Retrieved on        = 2011-05-12
> length(reactome)
[1] 1070
> nci[[1]]
"ALK1 pathway" pathway from NCI
Number of nodes     = 312
Number of edges     = 890
Type of identifiers = native
Retrieved on        = 2011-05-12
> length(nci)
[1] 177
>
```

```
>
> grep("Neuro",names(kegg))
[1] 129 130
> p<-kegg[[129]]
> p
"Neuroactive ligand-receptor interaction" pathway from KEGG
Number of nodes      = 272
Number of edges      = 46
Type of identifiers = native
Retrieved on         = 2011-05-05
> nodes(p)[1:10]
 [1] "EntrezGene:7442"   "EntrezGene:1141"   "EntrezGene:3363"
 [4] "EntrezGene:116443" "EntrezGene:8973"   "EntrezGene:2557"
 [7] "EntrezGene:6751"   "EntrezGene:3350"   "EntrezGene:5734"
[10] "EntrezGene:1136"
> edges(p)[1:10]
Error in `[.data.frame`(edges(p), 1:10) : undefined columns selected
Calls: [ -> [.data.frame
> edges(p)[1:10,]
                   src              dest direction       type
20649 EntrezGene:1081  EntrezGene:2492  directed activation
20650 EntrezGene:1081  EntrezGene:3973  directed activation
20651 EntrezGene:1081  EntrezGene:7253  directed activation
20652 EntrezGene:1442  EntrezGene:2690  directed activation
20653 EntrezGene:1511  EntrezGene:2149  directed activation
20654 EntrezGene:1511  EntrezGene:2150  directed activation
20655 EntrezGene:1511  EntrezGene:2151  directed activation
20656 EntrezGene:1511 EntrezGene:56288  directed activation
20657 EntrezGene:1511  EntrezGene:9002  directed activation
20658 EntrezGene:2147  EntrezGene:2149  directed activation
> psym<-convertIdentifiers(p,"symbol")
> nodes(psym)[1:10]
 [1] "TRPV1"  "CHRNB2" "HTR7"   "GRIN3A" "CHRNA6" "GABRA4" "SSTR1"  "HTR1A"
 [9] "PTGER4" "CHRNA3"
> edges(psym)[1:10,]
    src  dest direction       type
1   CGA  FSHR  directed activation
2   CGA LHCGR  directed activation
3   CGA  TSHR  directed activation
4  CSH1   GHR  directed activation
5  CTSG   F2R  directed activation
6  CTSG F2RL1  directed activation
7  CTSG F2RL2  directed activation
8  CTSG PARD3  directed activation
9  CTSG F2RL3  directed activation
10   F2   F2R  directed activation
>
```

solve {base}

Solve a System of Equations

Description

This generic function solves the equations `a %*% x = b` for x, where b can be either a vector or a matrix.

Usage

```
solve(a, b, ...)

## Default S3 method:
solve(a, b, tol, LINPACK = FALSE, ...)
```

Arguments

a	a square numeric or complex matrix containing the coefficients of the linear system.
b	a numeric or complex vector or matrix giving the right-hand side(s) of the linear system. If missing, b is taken to be an identity matrix and `solve` will retu
`tol`	the tolerance for detecting linear dependencies in the columns of a. If `LINPACK` is TRUE the default is `1e-7`, otherwise it is `.Machine$double.eps` .
`LINPACK`	logical. Should LINPACK be used (for compatibility with R < 1.7.0)? Otherwise LAPACK is used.
...	further arguments passed to or from other methods

Details

a or b can be complex, but this uses double complex arithmetic which might not be available on all platforms and LAPACK will always be used.

The row and column names of the result are taken from the column names of a and of b respectively. If b is missing the column names of the result are the row n

For back-compatibility a can be a (real) QR decomposition, although `qr.solve` should be called in that case. `qr.solve` can handle non-square systems.

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

See Also

`solve.qr` for the qr method, `chol2inv` for inverting from the Choleski factor, `backsolve`, `qr.solve`.

Examples

```
> 
> 
> 
> ?solve
> 
```

Search Results

The search string was**"ngs"**

Vignettes:

Biostrings::Biostrings2Classes	A short presentation of the basic classes defined in Biostrings 2
Biostrings::MultipleAlignments	Multiple Alignments
Biostrings::PairwiseAlignments	Pairwise Sequence Alignments

Code demonstrations:

base::is.things	Explore some properties of R objects and is.FOO() functions. Not for newbies!

Help pages:

affylmGUI::AboutaffylmGUI	Graphical User Interface for the limma microarray package
annotate::chromLocation-class	Class chromLocation, a class for describing genes and their chromosome mappings.
annotate::probesByLL	A function that does reverse the mappings between probe ids and the corresponding values
AnnotationDbi::toSQLStringSet	Convert a vector to a quoted string for use as a SQL value list
ath1121501.db::ath1121501ARACYC	Mappings between probe identifiers and KEGG pathway identifiers
ath1121501.db::ath1121501PATH	Mappings between probe identifiers and KEGG pathway identifiers
Biobase::strbreak	Break Character Strings to Fit Width
BiocGenerics::paste	Concatenate strings
Biostrings::class:AlignedXStringSet0	AlignedXStringSet and QualityAlignedXStringSet objects
Biostrings::class:SparseList	Biostrings internals
Biostrings::class:MultipleAlignment	MultipleAlignment objects
Biostrings::class:PreprocessedTB	PDict objects
Biostrings::class:PairwiseAlignedXStringSet	PairwiseAlignedXStringSet, PairwiseAlignedFixedSubject, and PairwiseAlignedFixedSubjectSummary objects
Biostrings::class:QualityScaledXStringSet	QualityScaledBStringSet, QualityScaledDNAStringSet, QualityScaledRNAStringSet and QualityScaledAAStringSet objects
Biostrings::class:XStringQuality	PhredQuality, SolexaQuality and IlluminaQuality objects
Biostrings::class:XStringSet	XStringSet objects

```
??ngs
```

Contributed Packages

Available Packages

Currently, the CRAN package repository features 4416 available packages.

Table of available packages, sorted by date of publication

Table of available packages, sorted by name

Installation of Packages

Please type help("INSTALL") or help("install.packages") in R for information on how to install packages from this repository. The manual R Installation and Administration [PDF] (also contained in the R base sources) explains the process in detail.

CRAN Task Views allow you to browse packages by topic and provide tools to automatically install all packages for special areas of interest. Currently, 30 views are available.

Package Check Results

All packages are tested regularly on machines running Debian GNU/Linux, Fedora and Solaris. Packages are also checked under Mac OS X and Windows, but typically only on the day the package appears on CRAN.

The results are summarized in the check summary (some timings are also available). Additional details for Windows checking and building can be found in the Windows check summary.

Writing Your Own Packages

The manual Writing R Extensions [PDF] (also contained in the R base sources) explains how to write new packages and how to contribute them to CRAN.

Repository Policies

The manual CRAN Repository Policy [PDF] describes the policies in place for the CRAN package repository.

About Bioconductor

Bioconductor provides tools for the analysis and comprehension of high-throughput genomic data. Bioconductor uses the R statistical programming language, and is open source and open development. It has two releases each year, 610 software packages, and an active user community. Bioconductor is also available as an Amazon Machine Image (AMI).

Use Bioconductor for...

Microarrays
Import Affymetrix, Illumina, Nimblegen, Agilent, and other platforms. Perform quality assessment, normalization, differential expression, clustering, classification, gene set enrichment, genetical genomics and other workflows for expression, exon, copy number, SNP, methylation and other assays. Access GEO, ArrayExpress, Biomart, UCSC, and other community resources.

Sequence Data
Import fasta, fastq, ELAND, MAQ, BWA, Bowtie, BAM, gff, bed, wig, and other sequence formats. Trim, transform, align, and manipulate sequences. Perform quality assessment, ChIP-seq, differential expression, RNA-seq, and other workflows. Access the Sequence Read Archive.

Transcription Factors
Find candidate binding sites for known transcription factors via sequence matching.

Recent Courses
Explore material from recent courses, including Intermediate R / Bioconductor for High Throughput Sequence Analysis (February 14-15, 2013).