SignatureClust: A Tool for Landmark Gene-guided Clustering

User Manual

September 2009

Contents

1 Introduction

Clustering is a popular data exploration technique widely used in microarray data analysis. Most conventional clustering algorithms, however, generate only one set of clusters independent of the biological context of the analysis. This is often inadequate to explore data from different biological perspectives and gain new insights. We propose a new clustering model that can generate multiple versions of different clusters from a single dataset, each of which highlights a different aspect of the given dataset.

Using our SigCalc algorithm, we transform the expression data matrix into a gene signature matrix. We then use standard clustering algorithms to cluster the genes based on these gene signatures. The clustering performed on gene signatures results in new gene associations that were not apparent when clustering with the original gene expression data. The difference is illustrated in Figure 1.1. The project homepage is at `http://infos.korea.ac.kr/SignatureClust.php`.

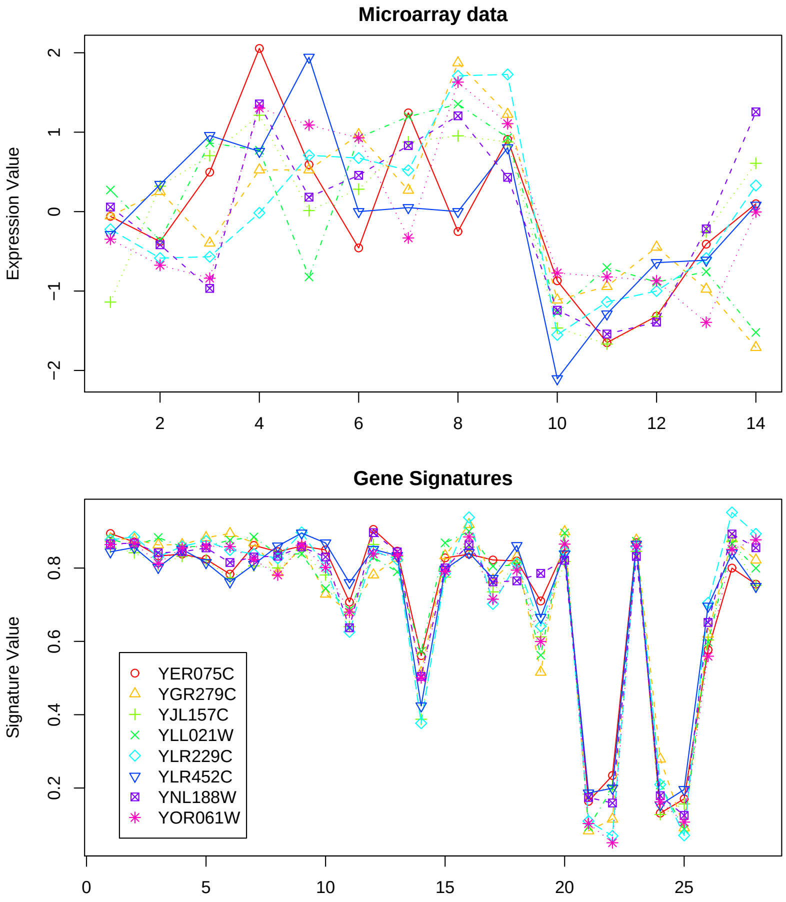


Figure 1.1: Comparison of microarray expression data with gene signatures for genes that clustered together using gene signatures. For the Gasch dataset (yeast), genes associated with *multi-organism process* (GO:0051704) were clustered together when landmarks associated with 'electron transport' were used.

2 Download & Installation

2.1 Downloading R and required packages

Download the R base binaries for Windows operating system from `http://cran.r-project.org/` and install these on your computer. After installing R and running the R GUI, download and install the graphviz in `http://www.graphviz.org/Download.php/`, which helps to make GO Term graph. At the moment of writing, the latest version of graphviz is 2.42. However, we strongly recommend to use graphviz 2.20 or earlier; the latest version fails to load some necessary DLLs to run the package such as libcdt-4.dll. Graphviz 2.20 can be obtained from `http://www.graphviz.org/pub/graphviz/stable/`.
Once installed, the path variable should be set. Go to `"Start>>Control Panel>> System>>Advanced [tab]>>Environment Variables>>System Variables"` and create new path variable, `GRAPHVIZ_INSTALL_DIR`, and set the value of it to `C:\Program Files\Graphviz 2.20`. Also, modify the pre-existing `PATH` variable to include `C:\Program Files\Graphviz 2.20\bin`. For example, if the original path was `C:\Program Files\R\R-2.8.1\bin`, it should be updated to `C:\Program Files\R\R-2.8.1 \bin;C:\Program Files\Graphviz 2.20\bin;`.
After all the base programs are installed, download the required R & Bioconductor packages by executing the following code:

```
source(“http://bioconductor.org/biocLite.R”)
biocLite(”Rgraphviz”)
biocLite(c(“GO.db”, “yeast2.db”, “KEGG.db”, “PFAM.db”, “goTools”, “GOstats”, “hgu133a.db”,
“hgu133b.db”, “hgu133plus2.db”, “hgu95av2.db”, “hgu95a.db”,
“hu6800.db”, “hgug4110b.db”, “hgug4112a.db”, “org.Hs.eg.db”,
“org.Mm.eg.db”, “org.Sc.sgd.db”))
```

Other R packages required for the application can be installed by executing:

install.packages(c(“cluster”, “som”, “fields”, “gWidgets”, “gWidgetstcltk”, “gWidgetsRGtk2”))

2.2 SignatureClust package

The SignatureClust.zip file consists of the following files and folders:

1. **‘DLLFiles’** folder - This contains the .dll files required for the application.
2. **‘sampleInputFiles’** folder - This contains sample microarray gene expression data files, and jpegs required for the application.
3. **‘SignatureClust.r’** file - This is the main R code file which has to be executed for the application to launch.
4. **‘functions.r’** file - This R file contains functions that are called from ‘SignatureClust.r’ file.

All of these files and folders are required for the application to run. On the first run of the application, two additional folders will be created:

1. **‘tempData’** folder - This will contain intermediate data files that are required by the tightclust algorithm.
2. **‘results’** folder - This is where all the output results will be stored. Dated sub-folders are created in this folder each time the SignatureClust application is executed.

2.3 Sample input data files

We have provided three microarray gene expression data files with the application. All the sample data files are located in the ‘sampleInputFiles’ folder. The details for these files are:

1. **dat_yeast.txt**: This has 2642 genes and 77 samples. For the yeast organism the gene names must be the gene symbols, as shown in the first column of this file.
2. **dat_hu6800.txt**: This has 450 genes and 31 samples. For Homo Sapiens, if the expression dataset is from one of the seven popular affymetrix or agilent arrays then probe ids can directly be used, as shown in the first column of this file.
3. **dat_SMD.txt**: This has 1709 genes and 15 samples. For Homo Sapiens, if the microarray is not one of the seven affymetrix or agilent arrays, then the first column of the data file must contain the Entrez IDs for the genes.

3 SignatureClust Application

3.1 Launch SignatureClust Application

After opening the R GUI, set the current working directory in R to the folder that contains the 'SigClust.r' file. For example, if the directory path to the downloaded 'SignatureClust' folder is `‘C:\Data\test\SigClust’`, then the command `setwd("C:/Data/test/SigClust")` would set the current directory to the folder containing the R files for SignatureClust. The application will launch when the following command is executed on the R GUI:

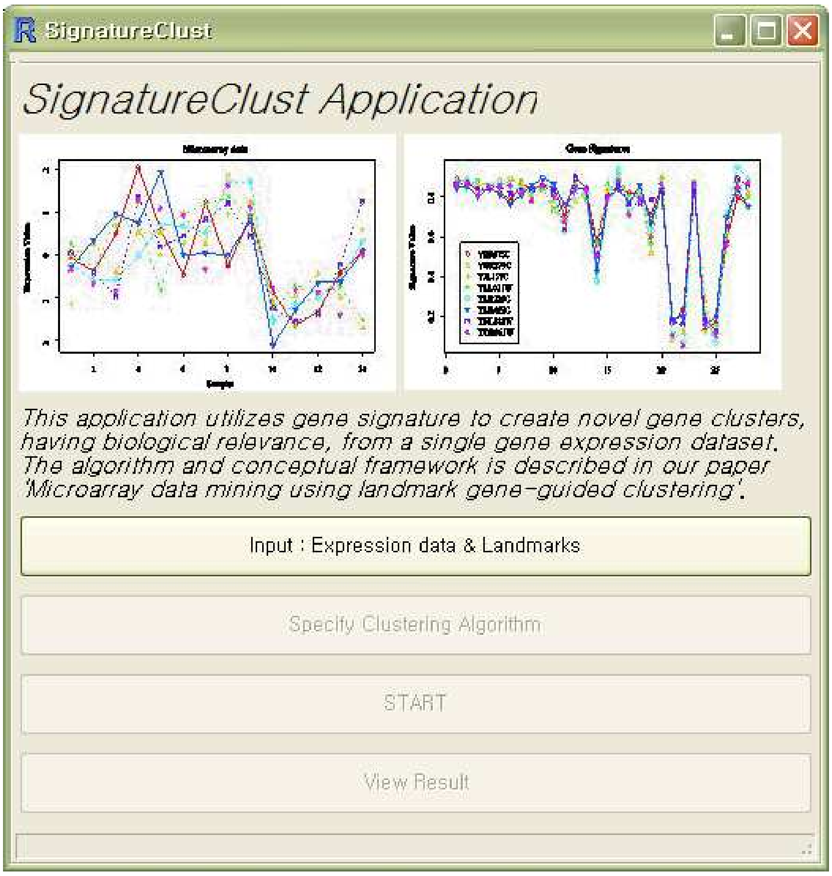
source("SigClust.r")

On execution, the application is launched. It has four buttons as shown in Figure 3.1. These represent the four steps required to get from raw data stage to the results stage. Initially, only the 'Input: Expression data & Landmarks' button is active. The other buttons become active when the preceeding data inputs have been completed (e.g. the 'Specify Clustering Algorithm' button will become active after data input has been completed in the section associated with the 'Input: Expression data & Landmarks' section). Clicking on this button will open the data input screen.

3.2 Input: Expression data and landmark genes (Step 1)

The Input data screen that appears will be similar to Figure 3.2.

1. ***Organism***: The organism for the gene expression dataset is selected. Currently only yeast and homo sapiens are supported by the application software. These are to be extended to zebrafish and rat. To follow the example that is


SignatureClust

SignatureClust Application

This application utilizes gene signature to create novel gene clusters, having biological relevance, from a single gene expression dataset. The algorithm and conceptual framework is described in our paper 'Microarray data mining using landmark gene-guided clustering'.

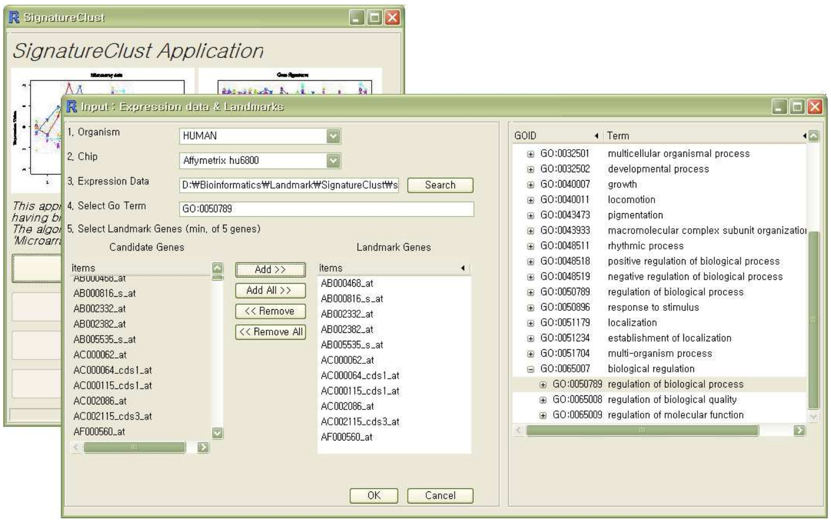
Input : Expression data & Landmarks
Specify Clustering Algorithm
START
View Result

Figure 3.1: SignatureClust application interface.

illustrated in the figures, select 'Human' from the drop down list.

2.***Chip***: In case of homo sapiens, if the chip is among the seven popular (affymetrix and agilent) chips shown, then the expression dataset can contain probe ids as rows (as shown in Figure 3.3). Otherwise, the first column must represent entrez ids. For example, if expression data is taken from Stanford Microarray Database (SMD), then the rows have to be entrez ids. In case of yeast, it is expected that the gene names are used in the rows (refer to sample yeast dataset provided). For the example, select 'Affymetrix hu6800' from the drop down list.

3.***Expression Data***: The path for the gene expression data is to be specified here. It is assumed that all pre-processing (e.g normalization, imputing missing values etc.) has already been done on the gene expression dataset. The input file should be a tab delimited text file (as shown in Figure 3.3). For



1. Organism: HUMAN

2. Chip: Affymetrix hu6800

3. Expression Data: D:₩Bioinformatics₩Landmark₩SignatureClust₩s

4. Select Go Term: GO:0050789

5. Select Landmark Genes (min. of 5 genes)

Candidate Genes

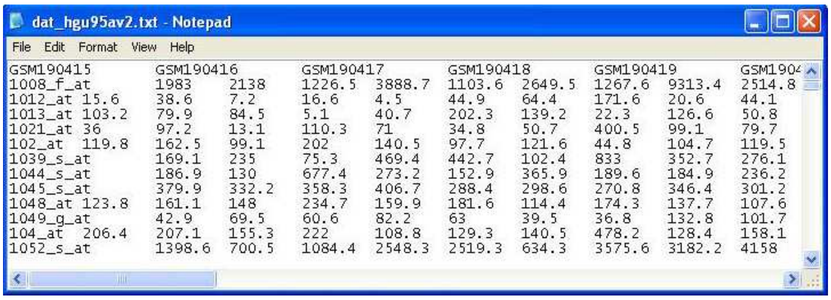
items
AB000468_at
AB000816_s_at
AB002332_at
AB002382_at
AB005535_s_at
AC000062_at
AC000064_cds1_at
AC000115_cds1_at
AC002086_at
AC002115_cds3_at
AF000560_at

Landmark Genes

items
AB000468_at
AB000816_s_at
AB002332_at
AB002382_at
AB005535_s_at
AC000062_at
AC000064_cds1_at
AC000115_cds1_at
AC002086_at
AC002115_cds3_at
AF000560_at

GOID	Term
⊞ GO:0032501	multicellular organismal process
⊞ GO:0032502	developmental process
⊞ GO:0040007	growth
⊞ GO:0040011	locomotion
⊞ GO:0043473	pigmentation
⊞ GO:0043933	macromolecular complex subunit organizatioı
⊞ GO:0048511	rhythmic process
⊞ GO:0048518	positive regulation of biological process
⊞ GO:0048519	negative regulation of biological process
⊞ GO:0050789	regulation of biological process
⊞ GO:0050896	response to stimulus
⊞ GO:0051179	localization
⊞ GO:0051234	establishment of localization
⊞ GO:0051704	multi-organism process
⊟ GO:0065007	biological regulation
⊞ GO:0050789	regulation of biological process
⊞ GO:0065008	regulation of biological quality
⊞ GO:0065009	regulation of molecular function

Figure 3.2: Input: Expression data and landmark genes.



```
GSM190415       GSM190416       GSM190417       GSM190418       GSM190419       GSM1904
1008_f_at       1983    2138    1226.5  3888.7  1103.6  2649.5  1267.6  9313.4  2514.8
1012_at 15.6    38.6    7.2     16.6    4.5     44.9    64.4    171.6   20.6    44.1
1013_at 103.2   79.9    84.5    5.1     40.7    202.3   139.2   22.3    126.6   50.8
1021_at 36      97.2    13.1    110.3   71      34.8    50.7    400.5   99.1    79.7
102_at  119.8   162.5   99.1    202     140.5   97.7    121.6   44.8    104.7   119.5
1039_s_at       169.1   235     75.3    469.4   442.7   102.4   833     352.7   276.1
1044_s_at       186.9   130     677.4   273.2   152.9   365.9   189.6   184.9   236.2
1045_s_at       379.9   332.2   358.3   406.7   288.4   298.6   270.8   346.4   301.2
1048_at 123.8   161.1   148     234.7   159.9   181.6   114.4   174.3   137.7   107.6
1049_g_at       42.9    69.5    60.6    82.2    63      39.5    36.8    132.8   101.7
104_at  206.4   207.1   155.3   222     108.8   129.3   140.5   478.2   128.4   158.1
1052_s_at       1398.6  700.5   1084.4  2548.3  2519.3  634.3   3575.6  3182.2  4158
```

Figure 3.3: Expression dataset for Affymetrix chip hgu95av2.

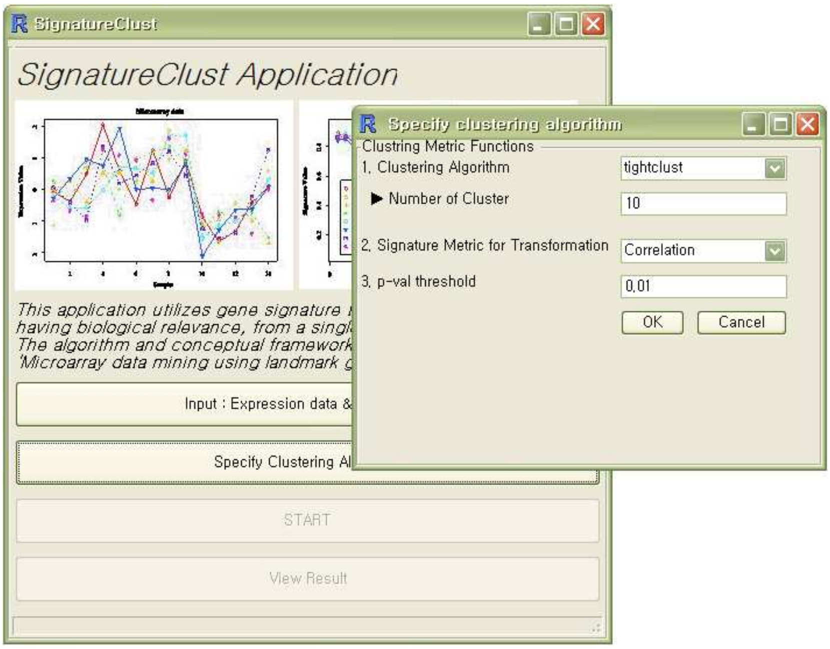
the example illustrated, navigate to the 'sampleInputFiles' directory and select `'dat_hu6800.txt'`.

4.***Select GO Term for landmark***: The screen initially has only one term specified ('biological process') with a small '+' mark next to it. By clicking on this, the first level of GO terms appears (this may take a few seconds). By clicking on the '+' symbols corresponding to each GO term, we are able to access it's subtree. By clicking on the '+' and '-' tabs, we are able to navigate through the Gene Ontology tree for biological process. For example, in Figure 3.2, the tree for the selected GO term is: 'biological process' *to* 'biological regulation' *to* 'regulation of biological processes'. Double click on the GO term (in this case 'regulation of biological processes') that you want, and all the probe ids/genes, in the input dataset, associated with this term will be listed under 'Candidate Genes'.

Alternatively, if you already know the GO term ID that you want as landmark, then you can directly enter the GO ID in the text box (for the example illustrated,‘GO:0050789’) and press ‘Enter’ on the keyboard. After a few seconds, the ‘Candidate genes’ box will be populated.

5.***Candidate Genes***: These are the probes/genes in the dataset that are associated with the selected GO term. To use all these probes/genes as landmarks, click on the ‘Add All’ button. Alternatively, users can select a subset of these genes to use as landmarks. A minimum of five genes must be selected for the SigCalc algorithm to function.

6.***Landmark Genes***: The genes selected as landmark genes will be displayed in this text box.



This application utilizes gene signature
having biological relevance, from a singl
The algorithm and conceptual framework
'Microarray data mining using landmark g

Clustring Metric Functions

1, Clustering Algorithm: tightclust

▶ Number of Cluster: 10

2, Signature Metric for Transformation: Correlation

3, p-val threshold: 0,01

Figure 3.4: Clustering input button becomes active.

Click on ‘OK’ to complete the Data Input section of the application. The main application menu will now show the ‘Specify Clustering Algorithm’ as active, as shown in Figure 3.4. Click on this to set the clustering algorithm parameters.

3.3 Specify Clustering Algorithm (Step 2)

1.***Clutering Algorithm***: Four popular clustering algorithms are provided in the application. These are tight clust, SOM, hierarchical clustering (diana) & Clara (Clustering Large Applications). The code for tightclust is courtesy of Mr George Tseng and a more detailed explanation of the algorithm can be found on his website:
`http://www.pitt.edu/~ctseng/research/tightClust_download.html`
The other three algorithms are R implementations and details can be found in the R documentation for these algorithms. Select the algorithm that you would like to use. Both tight clust and SOM are computationally intensive algorithms and the current implementation of them is suitable for datasets having less than 2000 genes. Usually, most gene expression analysis trims down the number of genes based on some criterion (e.g., variation of expression across samples) to a few thousand.
2.***Distance Metric for Clustering***: This specifies the distance metric that will be used by the clustering algorithm to form clusters.
3.***Signature Metric for Transformation***: This specifies the metric that will be used to create gene signatures from the gene expression matrix. Please refer to our paper [1] for details.
4.***P-val threshold***: This specifies the p-value threshold that the gene clusters must satisfy for the GO term to be considered as 'significant'. This is based on the hypergeometric test function of the GOstats package in bioconductor. A detailed explanation of this is given in our paper [1] [2], as well as in the R documentation for the GOstats package.
5.***N of Cluster***: The number of clusters desired.
6.***X dim and Y dim***: For the SOM algorithm, the x dimension, and the y dimension has to be specied to determine the number of clusters.

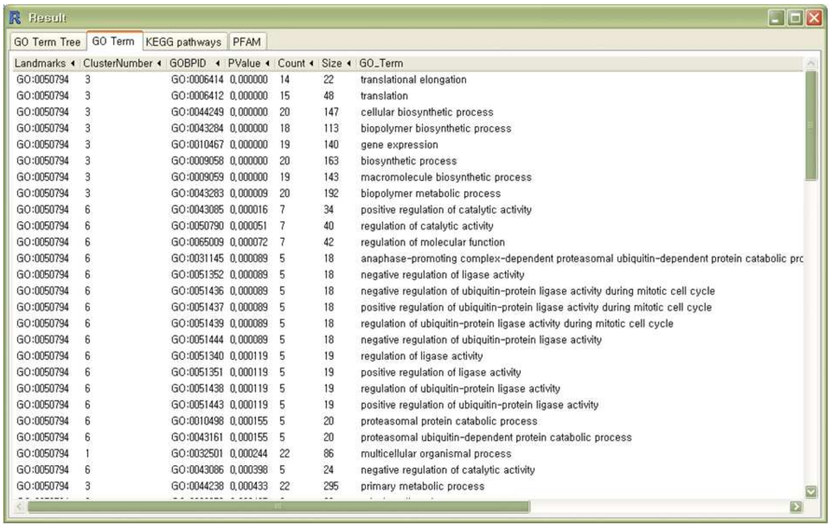
Click on 'OK' after entering the parameters for this section. For the example illustrated, all the default settings are accepted.

3.4 Start (Step 3)

After the input parameters have been set, click on the start button to begin execution. The execution time depends on the number of genes/probes in the expression dataset, the clustering algorithm chosen and the number of clusters. The progress of the program is given in the status bar appearing at the bottom of the SignatureClust application.

When the execution has completed, the 'View Results' button will become active.

3.5 View Results (Step 4)



Landmarks	ClusterNumber	GOBPID	PValue	Count	Size	GO_Term
GO:0050794	3	GO:0006414	0,000000	14	22	translational elongation
GO:0050794	3	GO:0006412	0,000000	15	48	translation
GO:0050794	3	GO:0044249	0,000000	20	147	cellular biosynthetic process
GO:0050794	3	GO:0043284	0,000000	18	113	biopolymer biosynthetic process
GO:0050794	3	GO:0010467	0,000000	19	140	gene expression
GO:0050794	3	GO:0009058	0,000000	20	163	biosynthetic process
GO:0050794	3	GO:0009059	0,000000	19	143	macromolecule biosynthetic process
GO:0050794	3	GO:0043283	0,000009	20	192	biopolymer metabolic process
GO:0050794	6	GO:0043085	0,000016	7	34	positive regulation of catalytic activity
GO:0050794	6	GO:0050790	0,000051	7	40	regulation of catalytic activity
GO:0050794	6	GO:0065009	0,000072	7	42	regulation of molecular function
GO:0050794	6	GO:0031145	0,000089	5	18	anaphase-promoting complex-dependent proteasomal ubiquitin-dependent protein catabolic prc
GO:0050794	6	GO:0051352	0,000089	5	18	negative regulation of ligase activity
GO:0050794	6	GO:0051436	0,000089	5	18	negative regulation of ubiquitin-protein ligase activity during mitotic cell cycle
GO:0050794	6	GO:0051437	0,000089	5	18	positive regulation of ubiquitin-protein ligase activity during mitotic cell cycle
GO:0050794	6	GO:0051439	0,000089	5	18	regulation of ubiquitin-protein ligase activity during mitotic cell cycle
GO:0050794	6	GO:0051444	0,000089	5	18	negative regulation of ubiquitin-protein ligase activity
GO:0050794	6	GO:0051340	0,000119	5	19	regulation of ligase activity
GO:0050794	6	GO:0051351	0,000119	5	19	positive regulation of ligase activity
GO:0050794	6	GO:0051438	0,000119	5	19	regulation of ubiquitin-protein ligase activity
GO:0050794	6	GO:0051443	0,000119	5	19	positive regulation of ubiquitin-protein ligase activity
GO:0050794	6	GO:0010498	0,000155	5	20	proteasomal protein catabolic process
GO:0050794	6	GO:0043161	0,000155	5	20	proteasomal ubiquitin-dependent protein catabolic process
GO:0050794	1	GO:0032501	0,000244	22	86	multicellular organismal process
GO:0050794	6	GO:0043086	0,000398	5	24	negative regulation of catalytic activity
GO:0050794	3	GO:0044238	0,000433	22	295	primary metabolic process

Figure 3.5: Unique significant GO terms revealed by signature clustering.

After execution of the program, results can be viewed by clicking this button. A tabbed screen appears that shows the results obtained for experiment. It has four tabs. The 'GO Term' tab shows unique Gene Ontology terms (highlighted) discovered using gene signature clustering that were not found by clustering of the original gene expression dataset. The other tabs list the significant KEGG/PFAM terms found. Figure 3.5 illustrates this. A detailed list of the significant GO terms and their associated p-values, alongwith a detailed list of genes in each cluster, is also output to the current working directory. The p-value signifies the probability associated with the genes in the cluster being grouped together by chance. The 'size' column indicates the number of genes in the input dataset that are mapped to that particular GO term. The 'count' column indicates the total number of genes in GO that are annotated to the term. The significant GO terms found are also mapped to the GO ontology tree (Figure 3.6) and results displayed in one of the tabs.

3.6 Output Files

Following the execution of the program, the following files are output to the 'results' directory.

1.***LandmarkSummary.out*** - This contains a summary of the results, i.e., the number of landmark genes used, the number of common and unique GO/KEGG/PFAM terms found by gene signature clustering.

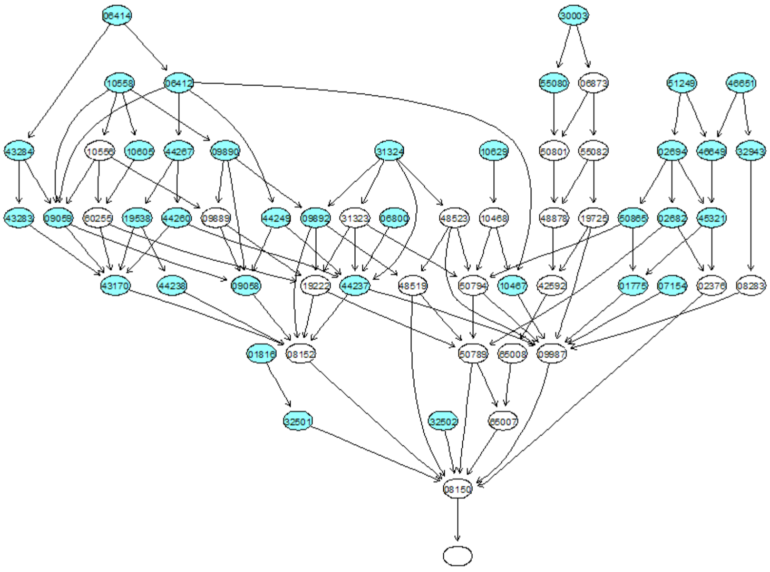


Figure 3.6: Unique significant GO terms mapped onto the GO ontology.

2.***geneSigClusters.out*** - This contains the actual clusters with gene/probe names. Each cluster is separated by '**' symbol.
3.***OriginalClustersXX.out*** - Where the 'XX' represents the clustering algorithm specified. This contains the gene/probe clusters using the original gene expression data.
4.***PvalsYY.out*** - Where the 'YY' represents the ontology (e.g GO, KEGG, PFAM). This contains the details of the unique GO/KEGG/PFAM terms found using gene signatures. It gives information on the GO/KEGG/PFAM ID, the associated p-value, the number of genes in the cluster that are associated with that GO/KEGG/PFAM term, and a description of the GO/KEGG/PFAM term. The graph in the 'View Results' section is derived from this file. A corresponding html file is also generated with hyperlinks to more information on the newly found GO/KEGG/PFAM terms.
5.***log.out*** - This contains a record of the clustering algorithm used, threshold p-value selected, dataset used etc.

3.7 Estimated Execution Times

For the three sample datasets provided with the application, we ran a short study to compare the execution times for different clustering algorithms on a Windows machine (3 GHz & 2 GB RAM). This should also aid the user in selecting an algorithm that would meet with their dataset requirements.

The graph for the execution times for the clustering algorithms, for varying datasets, is given in Figure 3.7

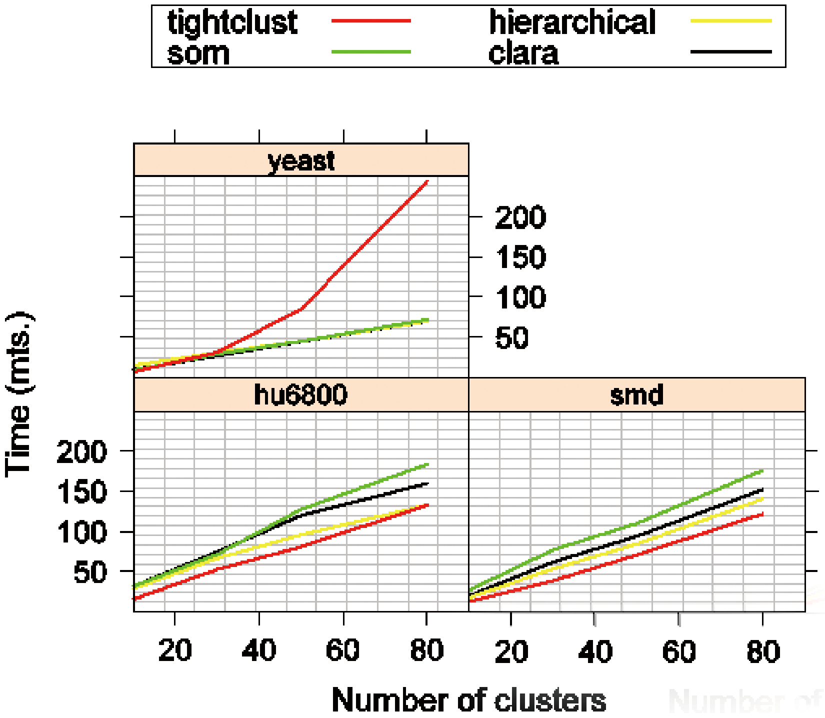


Figure 3.7: Time estimates for clustering algorithms for sample datasets.

4

Session Info for R

The sessionInfo() for R, at the time of creating this manual, is:

```
sessionInfo()
R version 2.8.1 (2008-12-22)
i386-pc-mingw32

locale:
LC_COLLATE=English_United States.1252;LC_CTYPE=English_United States.1252;
LC_MONETARY=English_United States.1252;LC_NUMERIC=C;
LC_TIME=English_United States.1252

attached base packages:
splines tools grid stats graphics grDevices utils

other attached packages:
gplots_2.6.0 gmodels_2.14.1 gdata_2.4.2
XML_1.99-0 biomaRt_1.16.0 affy_1.20.2
RankAggreg_0.3-1 gtools_2.5.0-1 gWidgetsRGtk2_0.0-50
fields_5.02 spam_0.15-3 GOstats_2.8.0
Category_2.8.4 genefilter_1.22.0 survival_2.34-1
RBGL_1.18.0 goTools_1.16.0 PFAM.db_2.2.5
KEGG.db_2.2.5 GO.db_2.2.5 org.Sc.sgd.db_2.2.6
org.Mm.eg.db_2.2.6 org.Hs.eg.db_2.2.6 hgug4112a.db_2.2.5
hgug4110b.db_2.2.5 hu6800.db_2.2.5 hgu95a.db_2.2.5
hgu95av2.db_2.2.5 hgu133plus2.db_2.2.5 hgu133b.db_2.2.5
hgu133a.db_2.2.5 som_0.3-4 yeast2.db_2.2.5
RSQLite_0.7-1 DBI_0.2-4 cluster_1.11.12
Rgraphviz_1.14.1 geneplotter_1.20.0 annotate_1.20.1
xtable_1.5-4 AnnotationDbi_1.4.3 lattice_0.17-20
```

Biobase_2.2.2 graph_1.20.0 gWidgets_0.0-32

loaded via a namespace (and not attached):
affyio_1.10.1 GSEABase_1.4.0 KernSmooth_2.22-22
MASS_7.2-45 preprocessCore_1.4.0 RColorBrewer_1.0-2
RCurl_0.94-0 RGtk2_2.12.9

Acknowledgements

The tightclust algorithm and code is courtesy Mr. George Tseng and can be found on his webpage at:
`http://www.pitt.edu/~ctseng/research/tightClust_download.html`

Bibliography

[1] Pankaj Chopra, Jaewoo Kang, Jiong Yang, Hyungjun Cho, Heenam S. Kim, and Min-Goo Lee. Microarray data mining using landmark gene-guided clustering. *BMC Bioinformatics*, 9:92+, February 2008.

[2] Pankaj Chopra, Hanjun Shin, and Jaewoo Kang. Microarray data mining using landmark gene-guided clustering. Submitted to *International Journal of Data Mining and Bioinformatics*, 2009.