TIGR MultipleExperimentViewer (MeV)

Overview

TIGR MultipleExperimentViewer (MeV) is an application that allows the user to view representations of processed microarray slides, and identify genes and expression patterns of interest. Slides can be viewed one at a time in detail or compared with each other. A variety of normalization algorithms and clustering analyses allow the user flexibility in creating meaningful views of the expression data. Reports or graphics can be created, containing the genes of interest.

Maintainer / Contact Person

TIGR MeV Team – mev@tigr.org

Platform / System Requirements

Java Runtime Environment (JRE) 1.3 or later
Java3D 1.2 or later required for PCA functions

Operation

- Create a preferences file. Preference file stores information about data input formats and database connection (if any). By using preference files, MeV can remain flexible and handle many input formats. Three sample preference files are included with the MeV installation, serving as templates for a .tav (TIGR ArrayViewer) file, Stanford file and cluster file. Of particular interest is the Additional Fields field, which allows any number of non-standard data fields to be read in and associated with each spot. Read the text of the preferences file for more information about the specific fields.

Fig 1. Preferences file.

- Run the batch file (TMEV.bat) to start the program. This batch file invokes the Java interpreter and stores input parameters.

- The Select Preference File dialog will be displayed, allowing you to choose a preferences file to start with. By default the dialog is pointed at the Preferences directory, so it is recommended that you store your preferences files there. A message will be written to the output window (created when MeV is started) if the preference file is valid.

  A preference file contains information about the format of the input file. Two major types of input files are supported: Stanford format, in which each file can contain data about multiple experiments, and the data are in the form of expression ratios; and tav (TIGR ArrayViewer) format, in which each file contains data from a single experiment, and Cy3 and Cy5 values are separately specified. Template preferences files for each type are provided with the software. Instructions for creating a preferences file are included in the preferences file itself. Open a copy of template file in a text editor, and make changes as necessary to customize the preference file for your input data file.

[redacted]

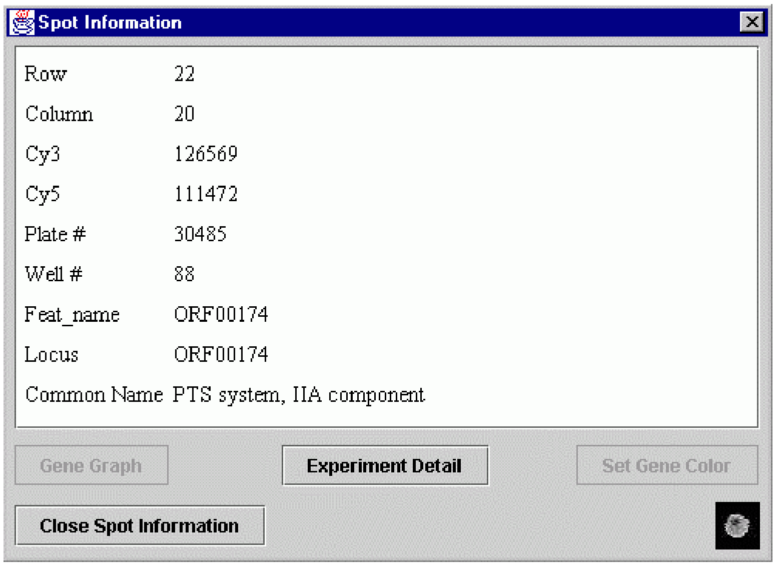
Fig 2. Preferences file chooser

- A main menu bar will appear, with four menus: File, Display, Window and References. Use the File menu in the main menu bar to open a new single or multiple array viewer, load a new preferences file or log in to a database. MeV will continue to run while this menu bar is present. To exit the entire application, select Quit from the File menu.

* A Single Array Viewer allows detailed views of one microarray slide at a time.

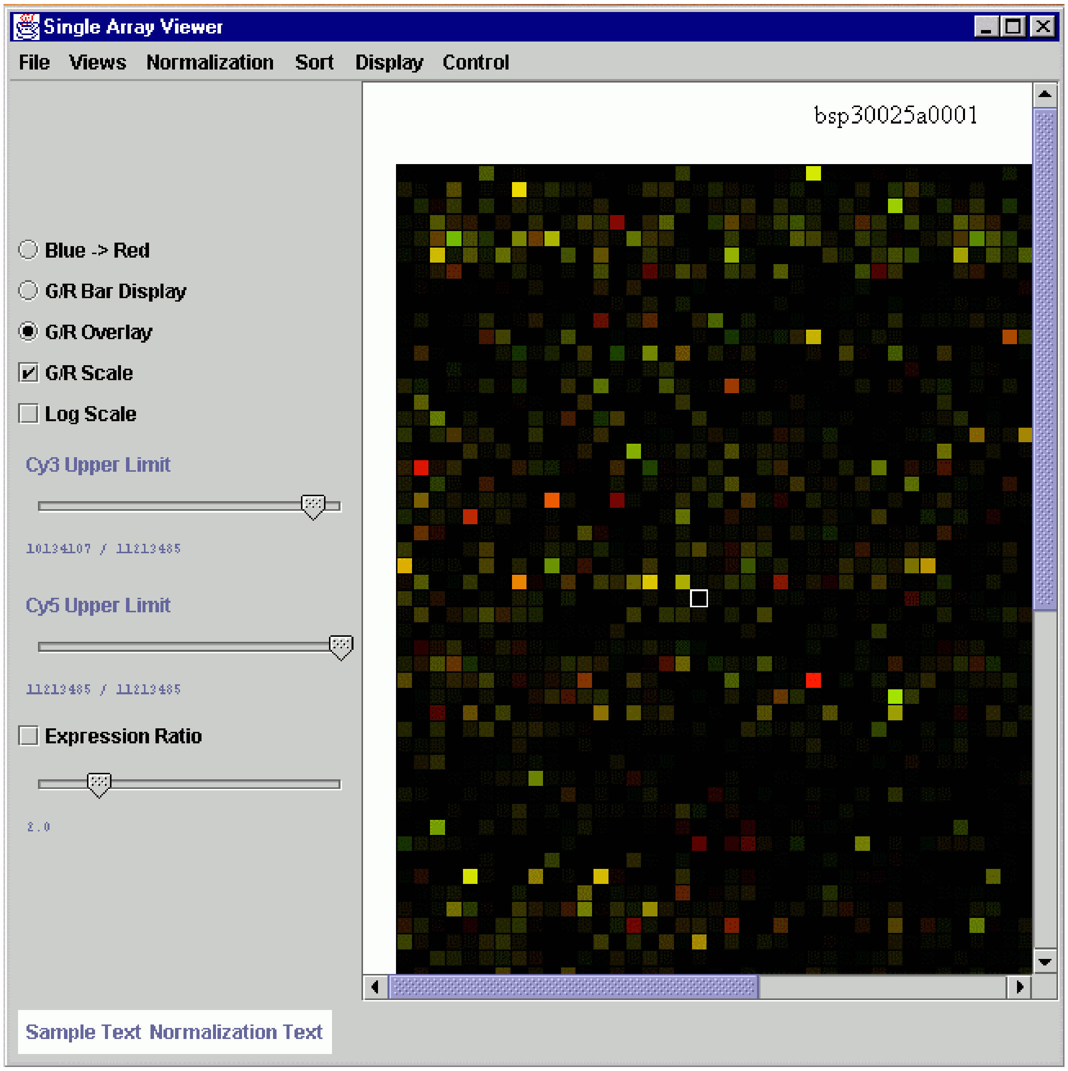
- Once a Single Array Viewer window is open, use its File menu (different from the main menu bar's File menu) to load a slide. Open Experiment From File loads a flat file (in .tav format) and Open Experiment From DB loads array data from a relational database using several stored procedures. If the user selects the former option, an open file dialog will be displayed, prompting the user to select a flat file to load. The latter option displays a list of experiment names and then analysis ids from the database. By selecting an experiment name (which represents a labeled slide) and analysis id (which represents a particular image analysis session) a unique dataset is specified.

- Once a slide has been loaded, a representation of the slide will be displayed in the window. Each colored rectangular bar (an element) corresponds to a spot on the array, and is in the same position. Clicking on a spot will display a dialog that shows detailed information about the target spot. This information includes the row and column of the spot, intensity values and the extra fields specified in the preference file. Other elements of this dialog include a compressed version of the actual spot image (where available) and a graph showing the expression levels of the gene across multiple experiments.



Row	22
Column	20
Cy3	126569
Cy5	111472
Plate #	30485
Well #	88
Feat_name	ORF00174
Locus	ORF00174
Common Name	PTS system, IIA component

Fig. 3. Spot information



○ Blue -> Red

○ G/R Bar Display

● G/R Overlay

☑ G/R Scale

☐ Log Scale

Cy3 Upper Limit

10134107 / 11213485

Cy5 Upper Limit

11213485 / 11213485

☐ Expression Ratio

2.0

Fig. 4. Single Array Viewer

- MEV allows four distinct normalization schemes and more can be added through the use of modules. One normalization can be selected at a time by using the Normalization menu. Each normalization scheme modifies the intensity values that were loaded, rather than changing the current values. Selecting No Normalization reverts the current values to the original ones.

- The default display is an overlay, showing the ratio between the two intensities and the absolute intensity levels. Greenish elements have a higher cy3 value, while reddish elements have a higher cy5 value. The brighter the element is, the higher the absolute levels of the intensities. The color scale is determined by the slider bars on the panel on the left side of the window. Lowering the values on the sliders makes the elements on the screen brighter. Other displays include a green/red bar display, and a false-color display. The green/red bar display divides each element with a vertical line showing the relative levels of the two intensities. An element that is two-thirds green has a cy3 value twice that of its cy5 value. The false-color display shows the two channels in separate areas, where elements are colored based on a scale where low intensities are dark and blue, while high intensities are bright and red. These displays can be selected using the panel on the left of the window.

- Several graphs can be created using the View Graph item in the Views menu. The choices include scatter plots of the two intensities, ratio histograms and a log ratio vs. log product graph. These graphs are displayed in a separate window, and mention the dataset and normalization scheme that spawned them.

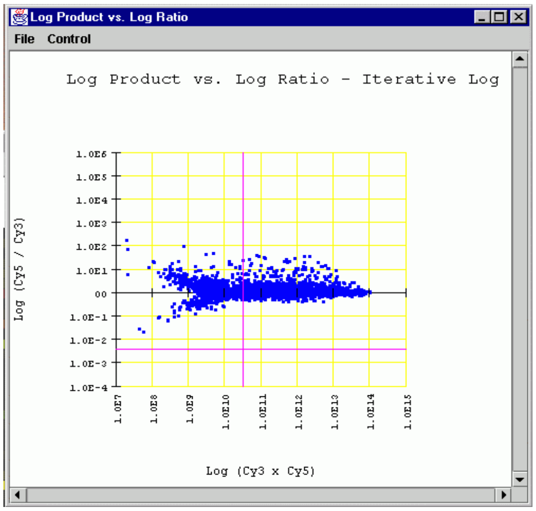


Fig. 5. Graph view from Single Array Viewer.

- The View Region item in the Views menu displays a dialog for inputting the coordinates of a metablock. A new Single Array Viewer is created, using the targeted metablock as the new dataset. In this way, the user can focus on a particular defined area of the slide.

- The elements of the array can be sorted by location (row and column), ratio or any of the additional fields that were specified in the preferences file. These sort options are available in the Sort menu. Sort by location is the default.

- Differentially expressed genes can be identified by checking the Expression Ratio checkbox on the panel on the left side of the window. The slider below the checkbox controls the expression ratio used to determine differential expression. When the checkbox is checked, only those genes which have one intensity value greater than the other by a factor greater than or equal to the expression ratio will be displayed. Other genes will be blacked out. For example, a ratio of 2.0 will exclude genes where the two intensities do not differ by a minimum factor of two.

- A sub array can be created to view just those genes that are still displayed after applying the expression ratio. Select the View Sub Array item from the Views menu to create a new Single Array Viewer window with the selected elements. These elements will be rearranged to eliminate gaps in the display.

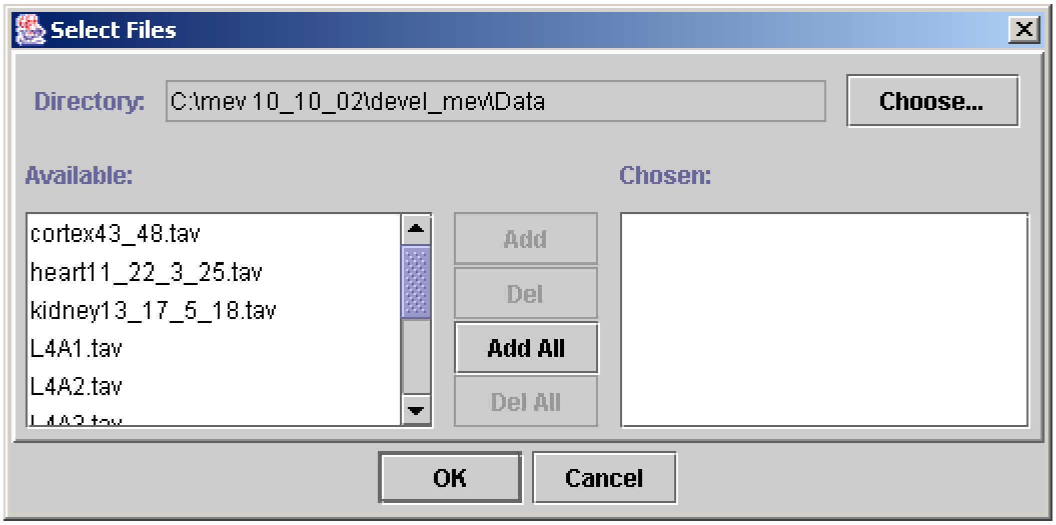
- To write a flat file as output, select the Generate Report item from the File menu. A save file dialog will be displayed, prompting for a name for the report file. This report will contain the data for each spot that is currently visible in a tab delimited format similar to the .tav format. The first lines of the report contain the name of the original input file and the normalization method used.

Fig. 6. Flat file output from Single Array Viewer

- The array representation can be saved as an image file or sent to a printer. Select Save Image from the File menu and choose a name and graphic format in the dialog that appears. To print the image, select Print Image from the File menu and set up the printer dialog.

* A Multiple Array Viewer allows comparisons of several experiments at once.

- Once a Multiple Array Viewer window is open, use its File menu to load any number of slides. Add Experiments from File opens a dialog where a .tav file can be loaded. Each loaded file will add a column to the display on the right side of the window. Add Experiments from Directory opens a dialog where all .tav files are listed on the left side, and selected .tav files for loading are on the right side. Files can be selected and deselected using the buttons. Once the desired files are in the list on the right side click the OK button to load all the select files. Add Experiment from DB loads an experiment from the database as in the Single Array Viewer. Add Experiments from Stanford File loads a file containing the data for multiple experiments (in Stanford format). Load Cluster displays an open file dialog that allows the user to specify a cluster saved in a previous MeV session.


Select Files

Directory: C:\mev 10_10_02\devel_mev\Data

Available:
cortex43_48.tav
heart11_22_3_25.tav
kidney13_17_5_18.tav
L4A1.tav
L4A2.tav
L4A3.tav

Chosen:

Fig. 7. Dialog box for choosing multiple files in Multiple Experiment Viewer

- With each slide that is loaded, a column is added to the main display. Each column represents a single experiment, and each row represents a gene. The names of the experiments are displayed vertically above each column, and any field of interest can be displayed to the right of each row. For best results, each experiment that is loaded should have the same elements in the same order. Clicking a spot displays a dialog with detailed information about the target spot.

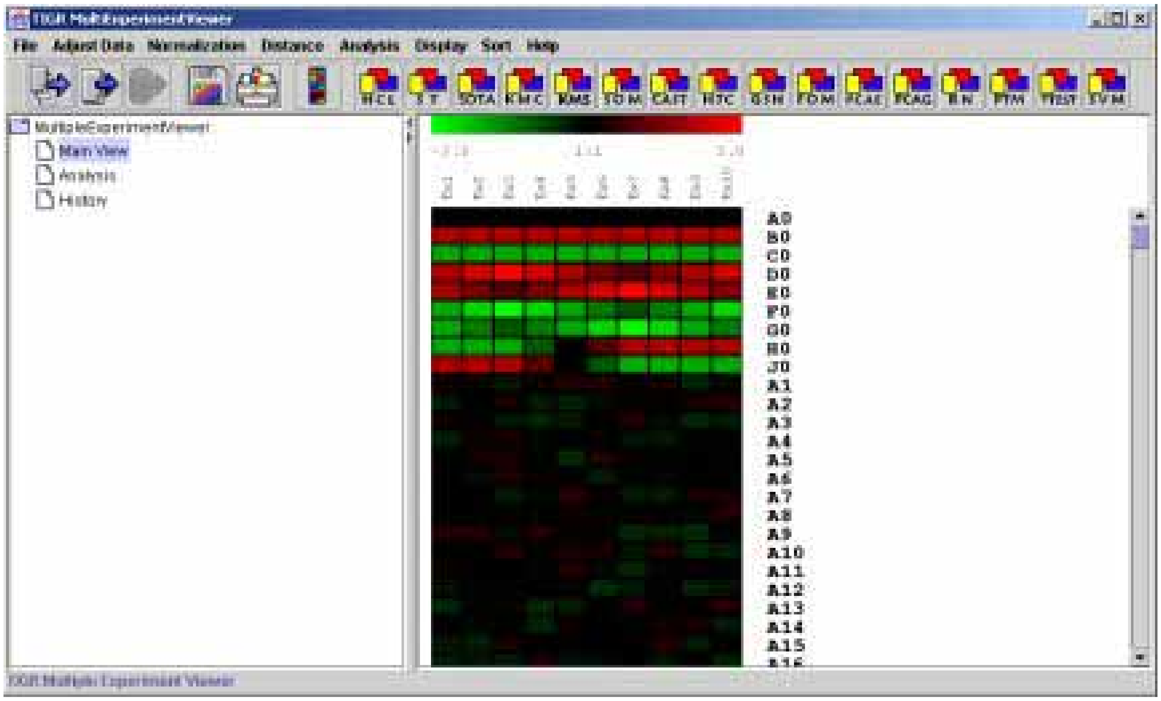


Fig. 8. Main View in MeV (with element borders drawn)

- The left side of the window is a navigation tree. At any time, clicking on the Main View node will return to the main display. As clustering analyses are performed, each one is added as a separate node in under Analysis. The History node lists all modifications (i.e., normalization and cutoffs) that have been made to the data.

- The Normalization menu offers the same methods as in Single Array Viewer. When a normalization method is selected, it is applied individually to each experiment.

- The default display is a green/red ratio display. Each element is red, green, black or gray. Black elements have a log ratio (cy5/cy3) of 0, while green elements have a log ratio less than 0, and red elements have a log ratio greater than 0. The further the ratio from 0, the brighter the element is. Gray elements have invalid values (Cy3 and Cy5 values are both zero) and are not used in any analyses. Other displays include a green/red bar display and the overlay display, both similar to the display modes in a Single Array Viewer window. The color scales for the overlay display can be set by using the Set Upper Limits item in the Display menu. These displays can be selected from the Display menu.

- Sorting based on location, ratio or additional fields can be performed using the items in the Sort menu. Each experiment is sorted individually. Sort by location is the default.

  CAUTION: When sorting is done based on ratio, the elements that make up a row in the main view no longer correspond to the same gene. As each column is sorted individually, sorting by ratio breaks up the association between columns.

- The labels to the right of each row can be modified or disabled using the Label item in the Display menu. Labels can be any of the additional fields specified in the preferences file.

- For all of the clustering analyses, it is necessary to select a distance metric. A distance metric can be chosen from the options in the Distance menu. The default distance metric varies depending on the analysis chosen.

- Prior to starting an analysis, certain data adjustments can be performed using the items in the Adjust Data menu. Some of these include normalization for experiments (where the experiments are normalized with each other), log transformations and various filters. Adjustments may not necessarily affect the main display, but will influence the calculation of the expression matrix, the foundation of all clustering analyses.

  CAUTION: Under the Adjust Data menu, the last option ("Adjust intensities of '0' ") is turned on BY DEFAULT. This means that if either (but not both) of the cy3 or the cy5 intensities for an element is recorded as zero, that intensity value will be reset to 1. In this case, the expression ratios will be calculated as cy5/1 or 1/cy3, depending on which value is zero, and the element is included in subsequent analyses. Sometimes, this may be desired

by the user. However, the user should be aware that the expression ratios for such elements are spurious. You might want to turn this option off, if you want to eliminate all those elements from the analysis that have at least one zero intensity value.

If both the cy3 and cy5 values are zero for an element, then that element is never included in the analysis.

- The expression matrix can be saved as a tab delimited text file by selecting the Save Matrix item from the File menu. Enter a name for the file in the save file dialog that is displayed.

Fig. 9. Expression matrix saved as text file.

- Most displays can be saved as an image file or sent to a printer. Select Save Image from the File menu and choose a name and graphic format in the dialog that appears. To print the image, select Print Image from the File menu and set up the printer dialog.

- **Hierarchical clustering** (**Eisen et al. 1998**) is available from the Analysis menu or the toolbar (**HCL**). Selecting this analysis will display a dialog that allows three different linkages and options to cluster genes, experiments or both. Once the computations are complete, select the Tree node under Analysis to view the hierarchical tree. The display is similar to the main display, but similar genes and experiments are connected by a series of 'branches'. Labels are displayed on the right side. Clicking near a branch intersection will select that branch and all branches to its right. Once selected, right clicking in the same area will display a popup menu that allows the user to set the highlighted area as a cluster, name the cluster, save the cluster and set several tree options. Clusters set and named in this display can propagate to other displays. Saving a cluster will display a dialog where a tab delimited text file containing the data for the highlighted cluster can be named.

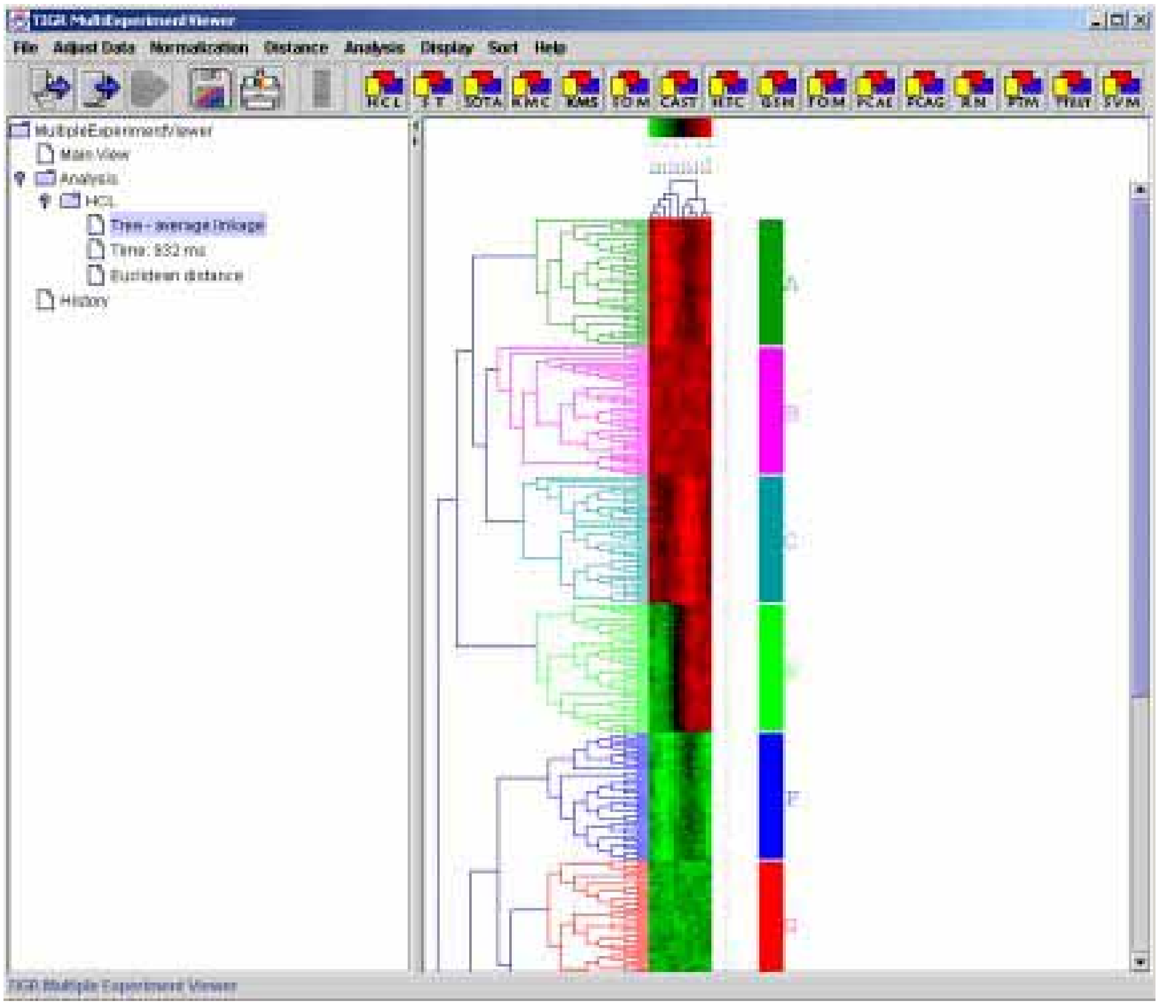


Fig. 10. Hierarchical tree with clusters selected.

- **Support Trees** are available from the Analysis menu or the toolbar (**ST**). This option shows the hierarchical trees obtained using the previous option, but it also shows the statistical support for the nodes of the trees, based on resampling the data. The user can select two resampling methods: bootstrapping (resampling with replacement), and jackknifing (resampling by leaving out one observation in this implementation). Resampling can be conducted on genes and / or experiments for a user-specified number of iterations. The branches of the resulting tree are color-coded to denote the percentage of times a given node was supported over the resampling trials. The legend for the color code corresponding to a given level of support can be found under the Help menu.

  The two most useful options for support trees are likely to be bootstrapping genes to build experiment trees, and bootstrapping experiments to build gene trees.

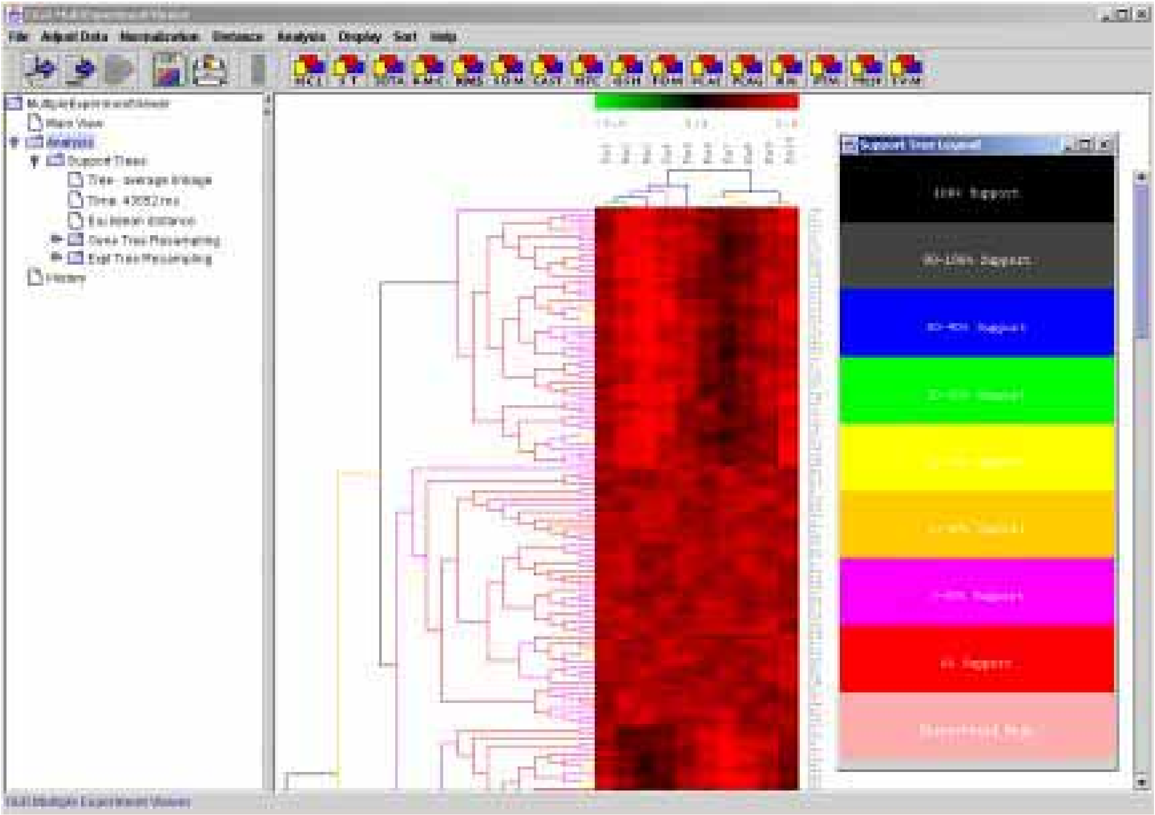


Fig. 11. Support trees for Hierarchical Clustering with Support Tree Legend displayed on right.

- **Self Organizing Tree Algorithm** (SOTA, **Dopazo et al. 1997, Herrero et al. 2001**) is available from the Analysis menu or the toolbar (**SOTA**). The initialization form shown below is divided into four main areas. The SOTA algorithm constructs a binary tree (dendrogram) in which the terminal nodes are the resulting clusters.

  The parameters roughly dictate the rules for growing the tree.

  1.) Growth Termination Criteria
  a.) Max Cycles – the maximum iterations allowed. The resulting number of clusters produced by SOTA is (Max Cycles +1).

  b.) Max epochs/cycle – the maximum number of training epochs allowed per cycle.

  c.) Max. Cell Diversity – all result clusters will fall below this diversity (mean gene to cluster centroid distance) if diversity is used as the cell division criteria. (Unless Max cycles are reached at which time some clusters may still exceed this parameter)

  d.) Min Epoch Error Improvement – describes the stability of cluster diversity required to indicate that training has reached stable state of a minimal cluster diversity.

e.) Run Maximum Number of Cycles (unrestricted growth) - SOTA runs until Max Cycles or until all of the input set are fully partitioned such that each cluster has one gene or several identical gene vectors.

2.) Centroid Migration and Neighborhood Parameters

a.) Migration Weights – are used to scale the movement of cluster centroids (characteristic gene expression patterns) toward a gene vector which has been associated with a neighborhood. When a gene is associated with a cluster the centroid adapts to become more like the newly associated gene vector. The parent and sister cell migration weights should be smaller than the weight for the winning cell (Cell to which the gene vector is associated.).

b.) Neighborhood Level – integer value from 0 to 5. This parameter dictates the resulting cell (terminal node) population which can receive associations from the training gene vector input set. Starting from the node containing the gene vectors to redistribute, the algorithm ascends the tree the indicated number of levels. Starting from this subroot all terminal nodes are candidates to receive new gene vectors from the current input set.

3.) Cell Division Criteria

a.) Use Cell Diversity – cell having the largest mean gene to centroid distance is the next cell to divide provided its diversity falls above Max. Cell Diversity (see 1c.).

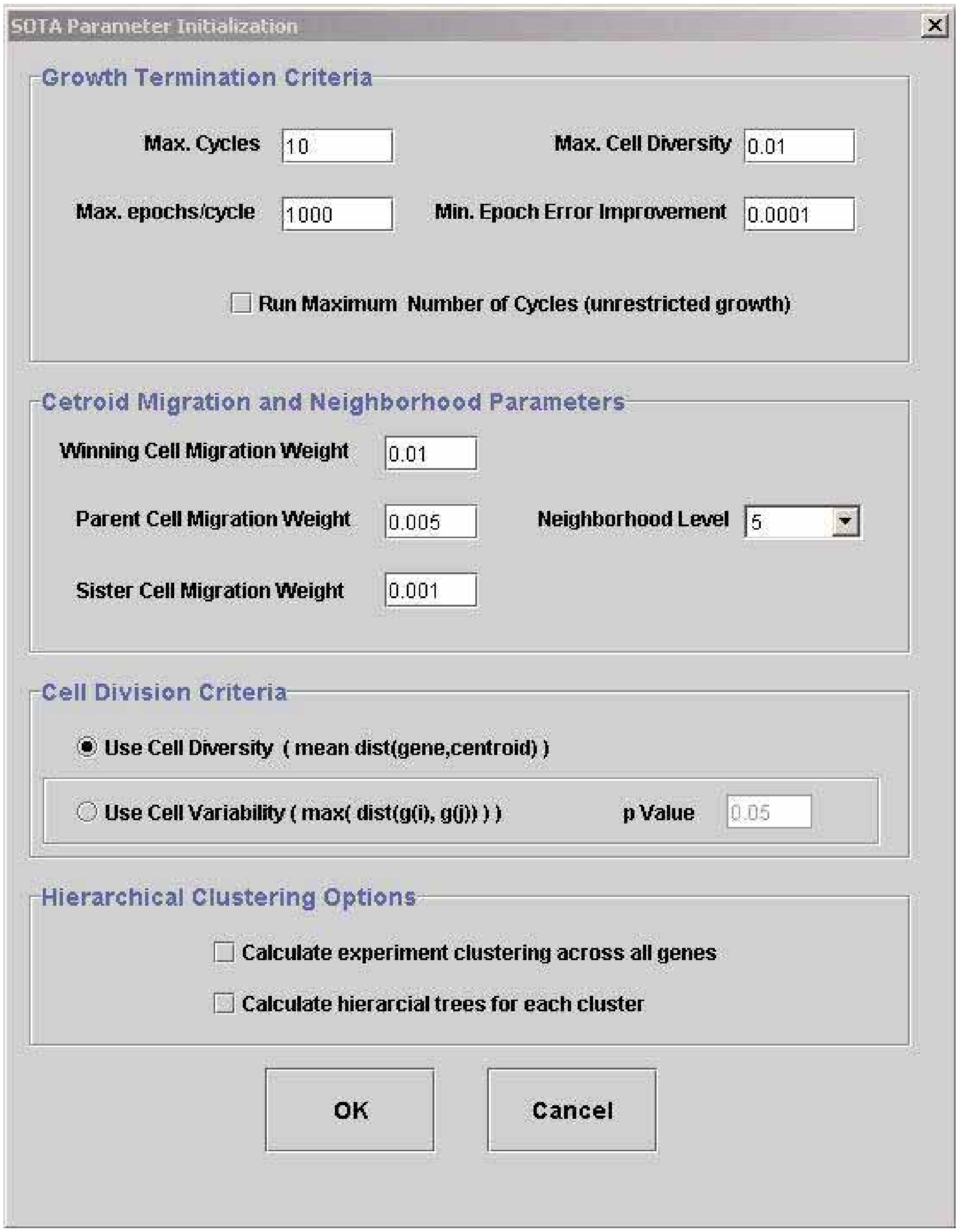
b.) Use Cell Variability – cell having the largest internal gene-to-gene distance is selected as the next cell to divide. Stopping criteria is changed so that growth continues until the most 'variable' cell falls below a variability criteria. (See below)

c.) pValue – used when using variability as the cell division criteria. A distribution of all gene to gene distances is generated by resampling the data set with each gene vector having randomized ordering of vector elements. The resulting distribution represents random gene to gene distances. The pValue supplied is applied to this resampled distribution to generate a variability cutoff. Clusters falling below this variability cutoff have a probability of having members that are paired by chance at or below the supplied pValue.

4.) Hierarchical Clustering Options

a.) Calculate experiment clustering across all genes – hierarchical clustering of experiments will be performed on the entire data set. Resulting tree is viewable over the SOTA dendrogram.

b.) Calculated hierarchical trees for each cluster – each resulting SOTA cluster will have its genes clustered and appropriate results will be displayed.



SOTA Parameter Initialization

Growth Termination Criteria

Max. Cycles 10

Max. Cell Diversity 0.01

Max. epochs/cycle 1000

Min. Epoch Error Improvement 0.0001

□ Run Maximum Number of Cycles (unrestricted growth)

Cetroid Migration and Neighborhood Parameters

Winning Cell Migration Weight 0.01

Parent Cell Migration Weight 0.005

Neighborhood Level 5

Sister Cell Migration Weight 0.001

Cell Division Criteria

● Use Cell Diversity (mean dist(gene,centroid))

○ Use Cell Variability (max(dist(g(i), g(j))))

p Value 0.05

Hierarchical Clustering Options

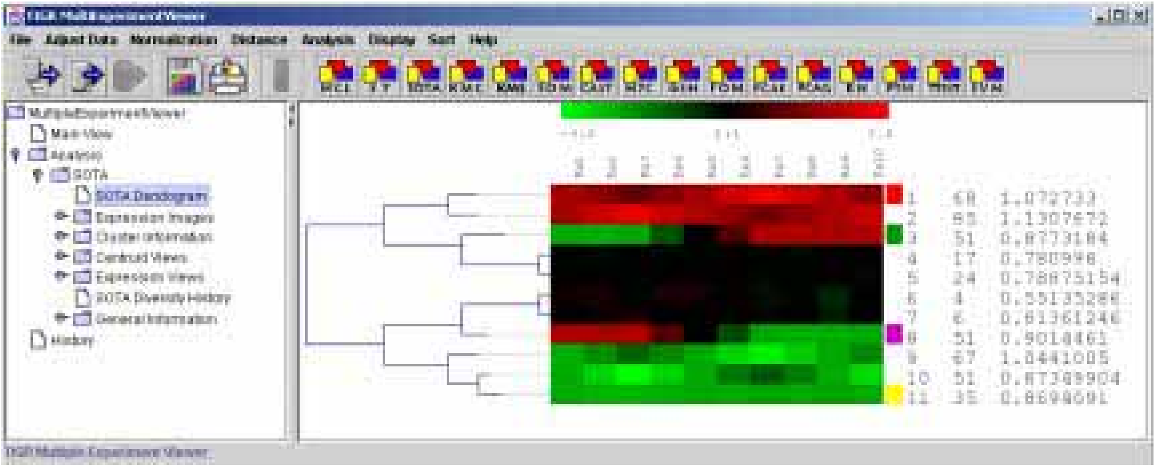
□ Calculate experiment clustering across all genes

□ Calculate hierarcial trees for each cluster

Fig 12. SOTA Initialization Dialog Box

The result views created by the SOTA algorithm are much like those created for k-means clustering (KMC) with the addition of two SOTA specific viewers and enhancements to expression image viewers to include more cluster information.

One of the SOTA specific viewers is the SOTA dendrogram (below) which displays the generated tree with the expression image of each resulting cluster's centroid gene. The text to the right of the centroid expression image includes a cluster id number, the cluster population (number of genes in the cluster), and the cluster diversity (mean gene to centroid distance). Clusters can be colored and saved from this viewer and a left click over a cluster centroid jumps to the expression image for that cluster.



```
1    68   1.072733
2    85   1.1307672
3    51   0.8773184
4    17   0.780998
5    24   0.78875154
6    4    0.55135286
7    6    0.81361246
8    51   0.9014461
9    67   1.0441005
10   51   0.87389904
11   35   0.8694691
```

Fig. 13. SOTA dendrogram.

The SOTA diversity viewer shows the change in the summation of gene to associated centroid distance vectors for all genes in the tree. This is a measure of overall tree diversity. This can reveal how much diversity improvement is achieved with each cycle (new cluster addition).

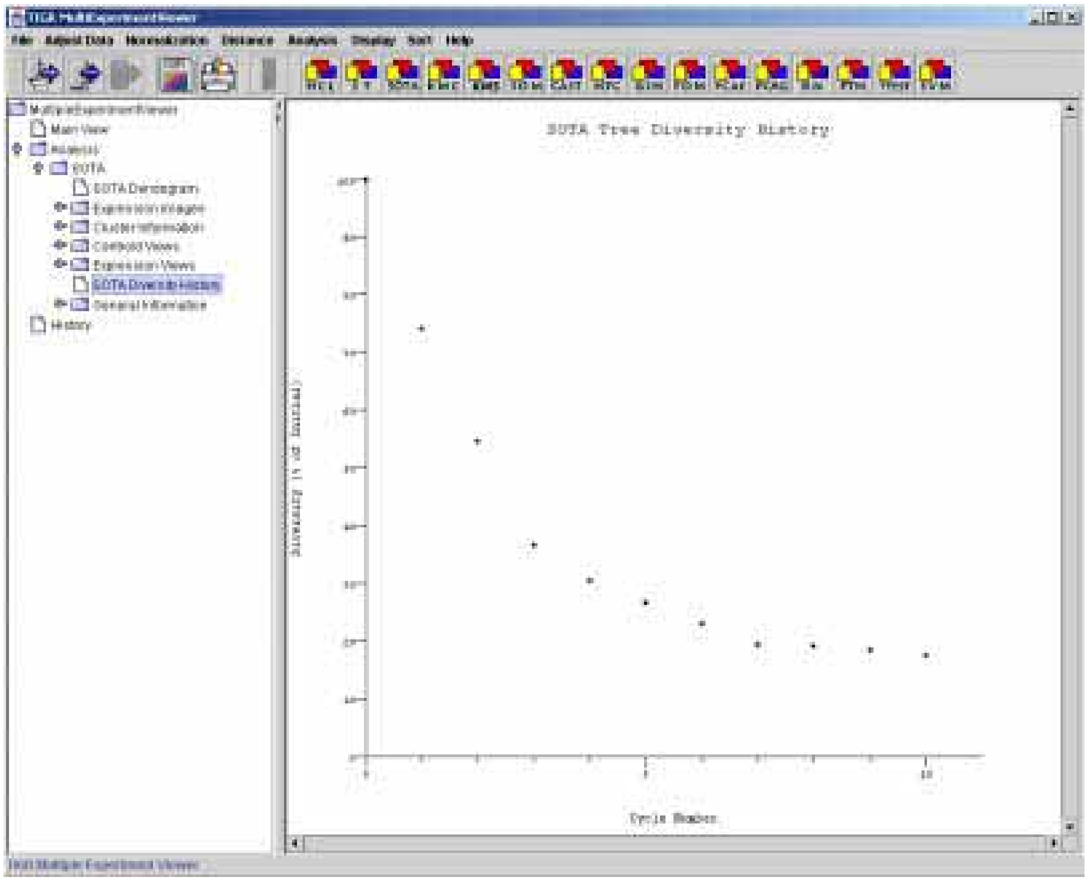


Fig. 14. SOTA diversity viewer

- **K-means/K-medians Clustering** (**Soukas et al. 2000**) is available from the Analysis menu or the toolbar (**KMC**). Selecting this analysis will display a dialog that allows the user to specify whether to use means or medians, as well as the number of clusters and iterations to run. Once the computations are complete, select the KMC node under Analysis to view the results. There are several subnodes beneath KMC, further divided by the clusters created based on the KMC input parameters. Hierarchical trees shows trees constructed for each cluster, if the option to draw hierarchical trees for clusters is selected. Expression images are similar to the main display. Cluster Information is a summary of each cluster based on size and % composition. Centroid views show the centroids for each cluster and experiment, individually or all at once. Expression views are similar to centroid views, but with each gene's expression levels displayed alongside the centroids. Right clicking within an expression image displays a popup menu that allows the user to propagate the cluster to other displays (Set Public Cluster), save and delete the cluster.

This method of clustering is useful when the user has an a priori hypothesis about the number of clusters that the genes should subdivide into.

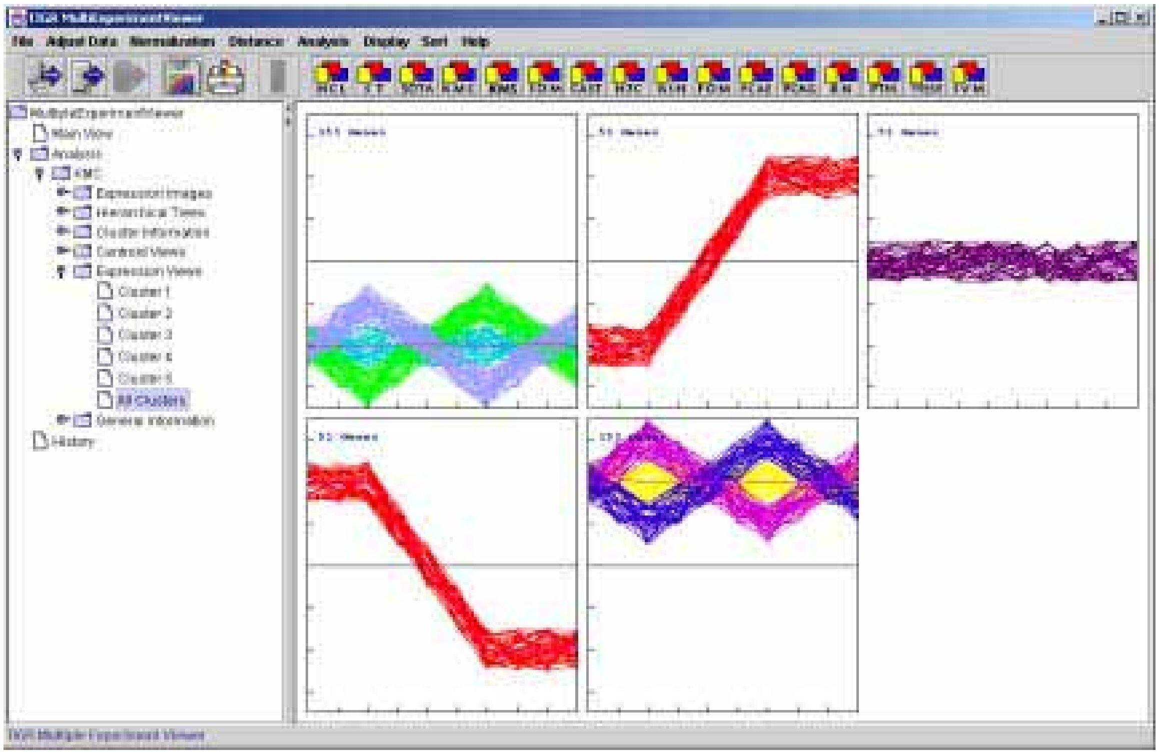
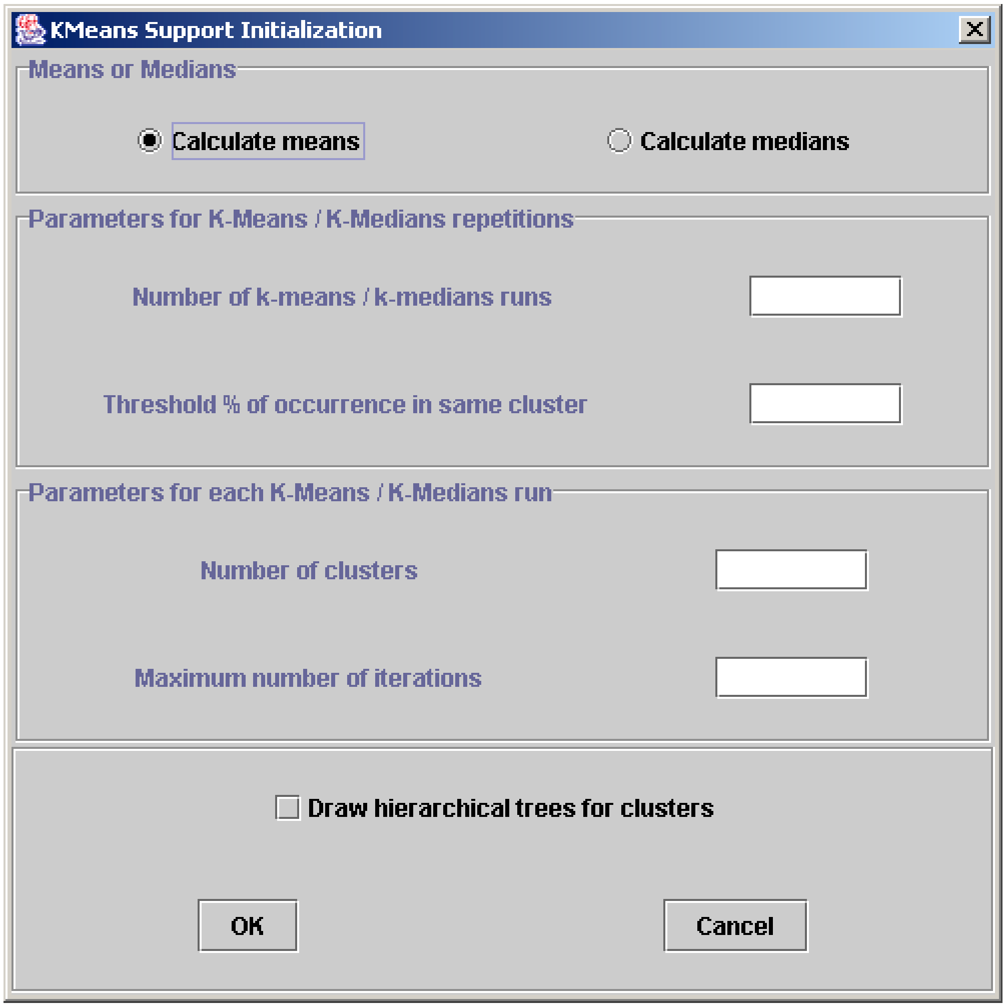


Fig. 15. K-Means / K-Medians Clustering: Expression Views.

- **K-Means / K-Medians Support** is available from the Analysis menu or the toolbar (**KMS**). This module allows the user to run the K-Means or K-Medians algorithms multiple times using the same parameters in each run. Owing to the random initialization of K-Means and K-Medians, the clusters produced may vary substantially between runs, depending on the data set and the input parameters. The KMS module allows the user to generate clusters of genes that frequently group together in the same clusters ("consensus clusters") across multiple runs. The output consists of consensus clusters in which all the member genes clustered together in at least *x*% of the K-Means / Medians runs, where *x* is the threshold percentage input by the user (see screenshot below).



Means or Medians

● Calculate means ○ Calculate medians

Parameters for K-Means / K-Medians repetitions

Number of k-means / k-medians runs

Threshold % of occurrence in same cluster

Parameters for each K-Means / K-Medians run

Number of clusters

Maximum number of iterations

□ Draw hierarchical trees for clusters

Fig. 16. K-Means / K-Medians Support: Initialization Dialog Box.

The number of consensus clusters generated may be more than the input number of clusters per run. This is because some genes may cluster together frequently, yet they may form a subset of different clusters in different runs. Hence, a set of genes that appeared as a single cluster in any given run may be split up into two or more consensus clusters over several runs. Some genes may remain unassigned because they did not cluster with any other genes in enough runs to exceed the threshold percentage.

- **Self Organizing Maps** (**Tamayo et al. 1999**) are available from the Analysis menu or the toolbar (**SOM**). Selecting this analysis will display a dialog that allows the user to set up the size, topology and behavior of the SOM. Once the computations are complete, select the SOM node under Analysis to view the SOM results. The subnodes under this node are very similar in form and function to those found beneath the KMC node.

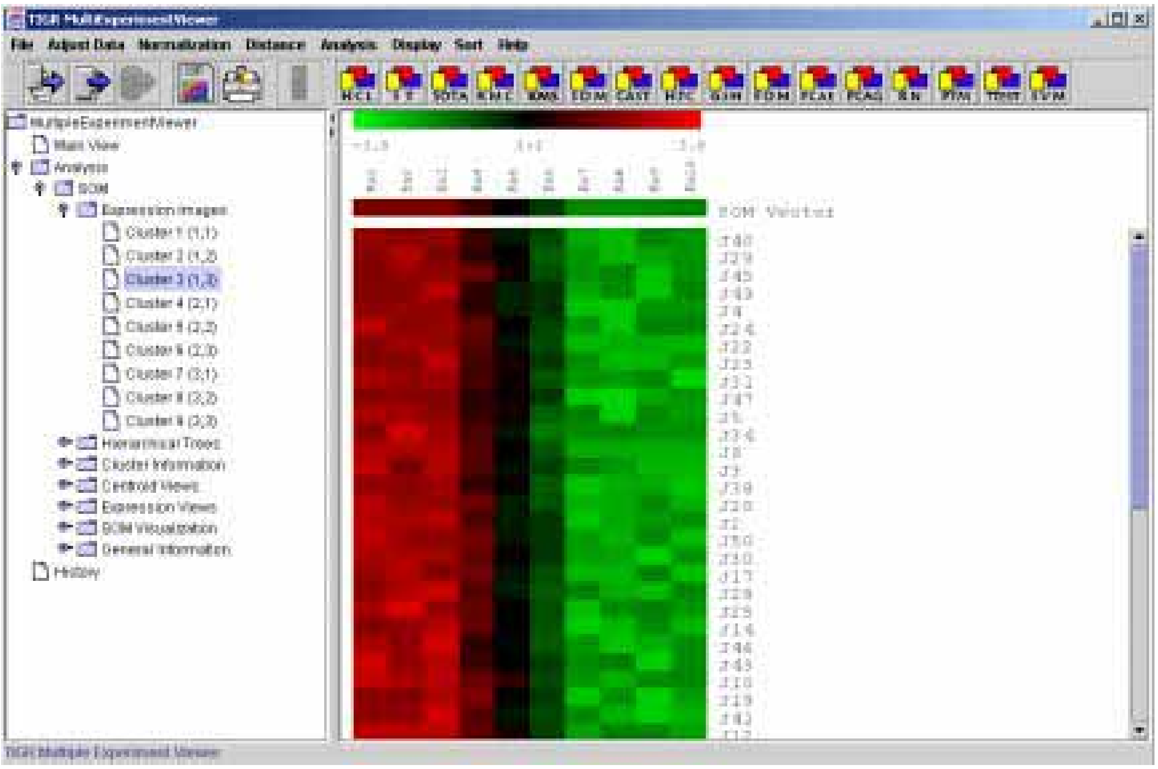


Fig. 17. SOM: Expression Image.

- The **Clustering Affinity Search Technique** (CAST, **Ben-Dor et al. 1999**) is available from the Analysis menu (Calculate CAST) or the toolbar (**CAST**). The user is prompted for a threshold affinity value between 0 and 1 (which may be thought of as the reciprocal of the distance metric between two genes, scaled between 0 and 1), that has to be exceeded by all genes within a cluster. The algorithm works by both adding and removing genes from a cluster, each time adjusting the affinities of the genes to the current cluster, and continuing this process until no further changes can be made to the current cluster.

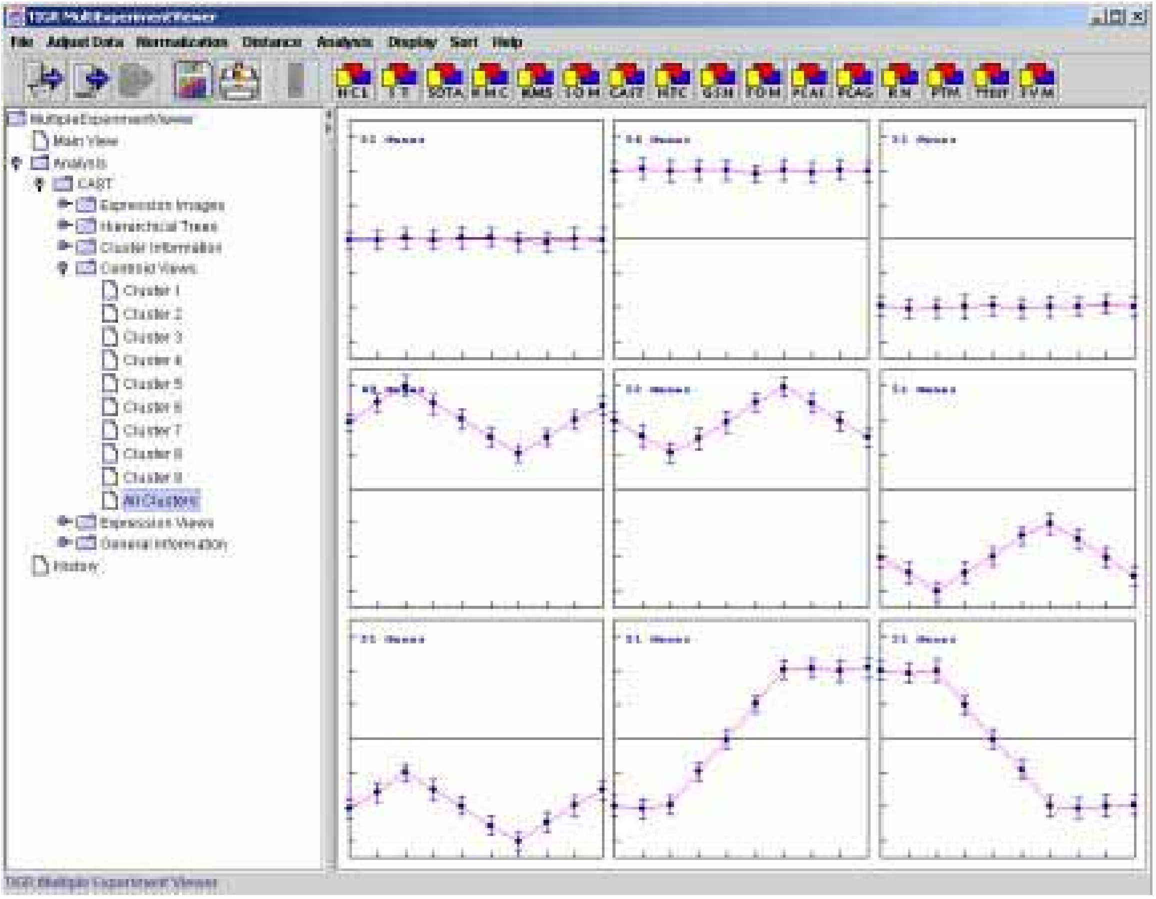
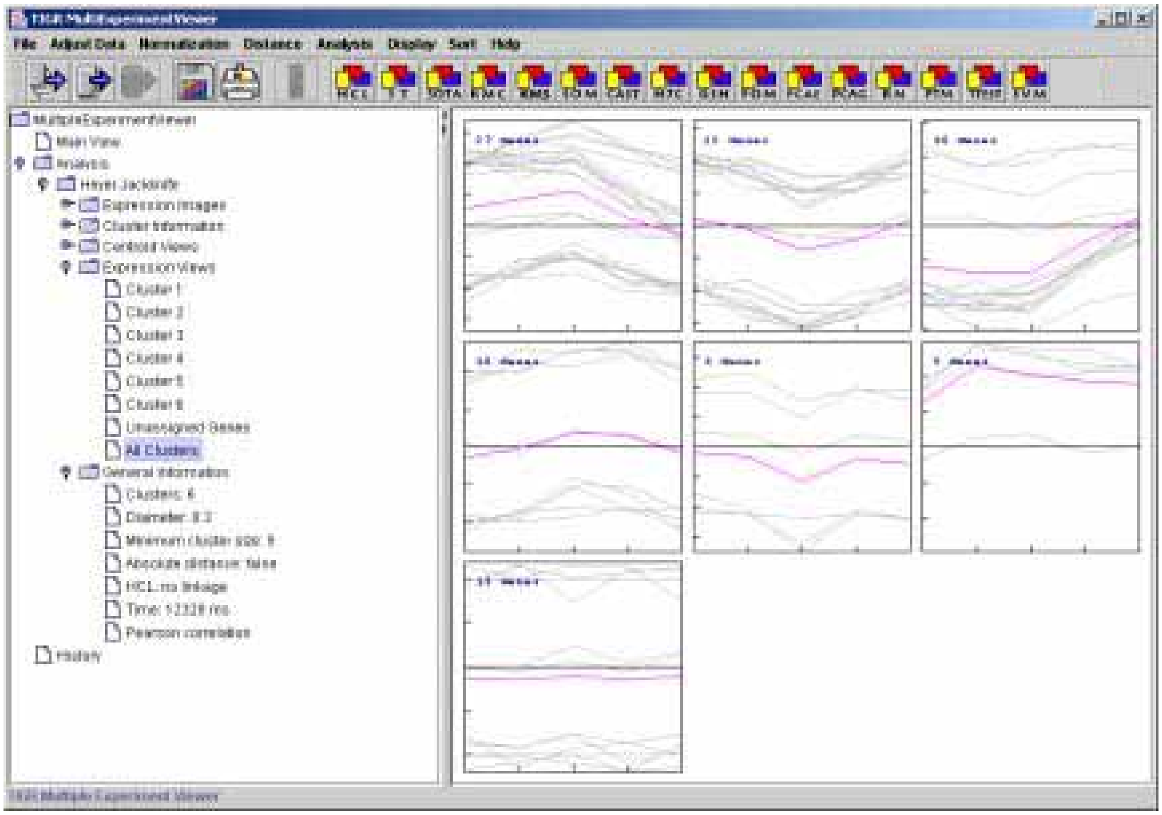


Fig. 18. CAST: Centroid View.

- ***QT CLUST*** (modified from **Heyer et al. 1999**) is available from the Analysis menu or the toolbar (**QTC**). The dialog will prompt the user for the cluster diameter and the minimum cluster size. The **cluster diameter** is the largest distance allowed between two genes in a cluster, expressed as a fraction between 0 and 1. A diameter of 1 corresponds to the largest possible distance between two genes. For Pearson Correlation, Pearson Uncentered, Pearson Correlation Squared, Kendall's Tau and Cosine Correlation, the maximum possible distance is 1, and therefore the user-input diameter is the actual maximum allowed distance between two genes in a cluster. For the other distance metrics, which do not have a fixed upper bound, the maximum distance between two genes in the current dataset is set to 1, and the diameter is calculated accordingly. To reduce bias resulting from outliers, the distances used for computing clusters are jackknifed, i.e., each experiment is left out in turn while computing the distance between two genes, and the maximum of the distances is taken. The **minimum cluster size** specifies the stopping criterion for the algorithm as it searches through the data set finding smaller and smaller clusters with each iteration. Checking the **Use Absolute** checkbox will include genes with similar as well as opposing trends in a cluster (e.g., if the distance metric selected is Pearson Correlation, both positively and negatively correlated genes will be considered for inclusion in a cluster). If the **Use Absolute** box is unchecked, only genes of similar trends will be considered for inclusion in the same cluster. As with the previous two methods, it is possible to construct hierarchical trees from the clusters. The last displayed group of genes consists of genes that remain unassigned to any cluster. The subnodes on the left panel are similar to the ones previously described for k-means and SOM.

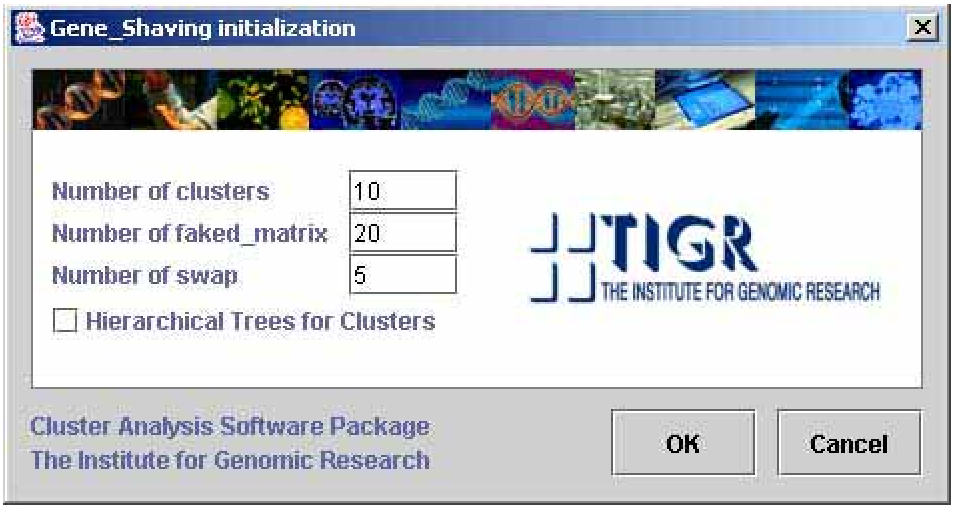


Clusters: 6
Diameter: 0.3
Minimum cluster size: 5
Absolute distance: false
HCL: no linkage
Time: 12328 ms
Pearson correlation

Fig. 19. QTC: Expression View.

- **GSH: Gene Shaving**
  (Hastie *et al.* 2000)

  The clusters that are created by this method differ from the results of other clustering algorithms in several ways. Clusters are constructed such that they show a large variation across the set of samples and small variation between the expression levels of the individual genes. Each cluster is independent of the others and they may overlap other clusters; each gene may belong to several clusters or none at all. One particularly interesting feature of this algorithm is that it will associate genes whose expression levels change by a similar magnitude across experiments, but in the opposite direction. For example, a gene with a given expression pattern across a series of experiments will be clustered with other genes whose expression pattern is the exact opposite.


Gene_Shaving initialization
TIGR
THE INSTITUTE FOR GENOMIC RESEARCH
Cluster Analysis Software Package
The Institute for Genomic Research
OK
Cancel

Number of clusters 10
Number of faked_matrix 20
Number of swap 5
□ Hierarchical Trees for Clusters

Fig. 20. Gene Shaving initialization dialog.

Parameters:

- Number of clusters:
  Number of gene clusters to be created.
- Number of faked_matrix:
  Number of randomized matrices used to calculate the sizes of the clusters.
- Number of swap:
  The number of times the expression matrix is permuted in order to create each randomized matrix.
- Hierarchical Trees for Clusters:
  If checked, hierarchical structures will be drawn for each cluster, describing the relationships between genes within that cluster.

Default Distance Metric: Euclidean

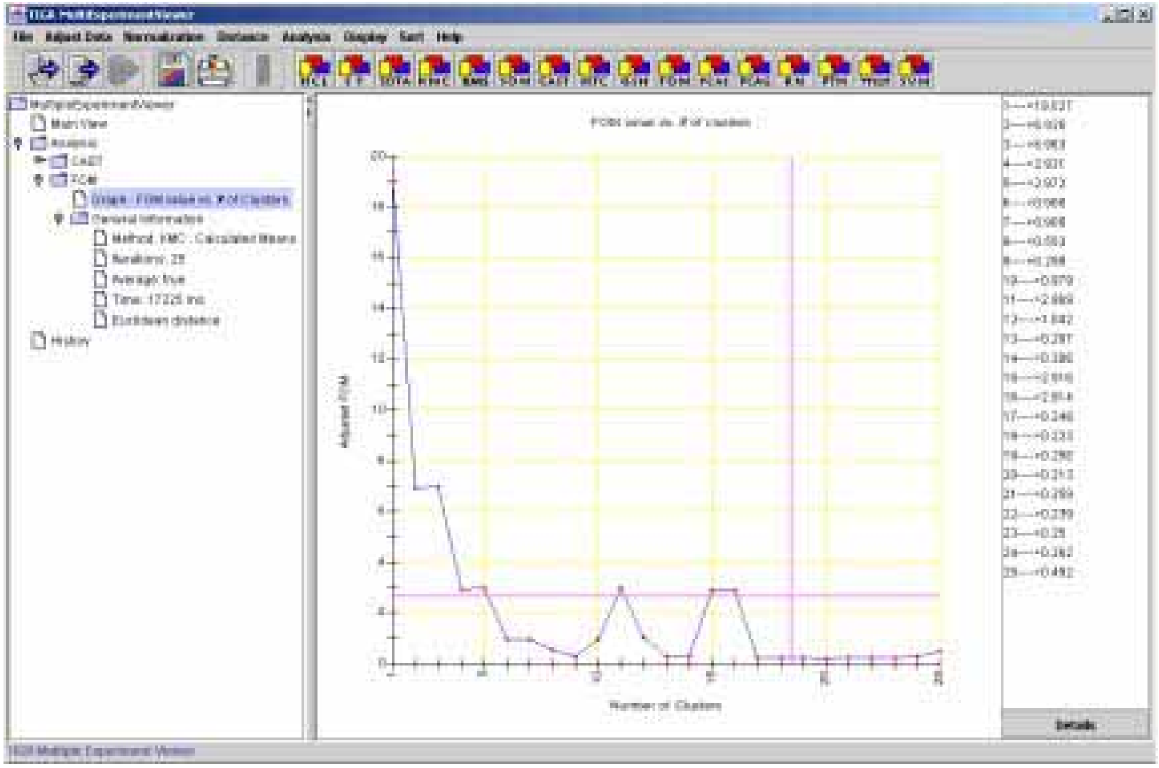
The first principle component of the expression matrix is calculated, and those genes whose variance is in the bottom 10% are cut out. These two steps are repeated until only one gene remains. This results in a series of nested clusters. One cluster is chosen from this series using the gap statistic (see below for details). The expression matrix is then orthogonalized, another series of nested clusters generated and one cluster chosen until the number of chosen clusters reaches the number specified in the “Number of clusters” parameter.

The method used to select one cluster out of a nested series is maximization of the Gap Statistic. Randomized clusters are created from the existing expression matrix. The ratio of expression variance of a given gene between experiments versus the variance of each gene about the cluster average is calculated. The cluster whose ratio is furthest from the average ratio of the randomized matrices is chosen.

This module can be slow, so it may be several minutes before results are displayed. The experiment subtree created by the module contains expression images, centroid views and expression views of each of the clusters predicted and the genes not assigned to clusters. It also contains a *Cluster Information* tab which reports the sizes of each cluster.

- **Figures of Merit** (**Yeung et al. 2001**) are available from the Analysis Menu (Generate FOM Graph) or the toolbar (**FOM**). Currently, FOM is available for the CAST, K-means and K-medians algorithms. A figure of merit is an estimate of the predictive power of a clustering algorithm. It is computed by removing each experiment in turn from the data set, clustering genes based on the remaining data, and calculating the fit of the withheld experiment to the clustering pattern obtained from the other experiments. The lower the adjusted FOM value, the higher the predictive power of the algorithm. The "Maximum number of clusters" input field under the K-Means / K-Medians tab in the initialization box is used to determine how many times FOM values should be calculated for the k-means/k-medians algorithm. Each time, the number of clusters computed is increased by one, starting with one cluster in the first iteration. The "Interval" input field under the CAST tab allows the user to specify the increase in threshold affinity in successive iterations of CAST. If the "Take Average" box is checked, in case there is more than one clustering outcome for a given number of clusters, the average FOM for that number of clusters will be used to draw the FOM-vs.-number of clusters curve. If the "Take Average" box is unchecked, each FOM value will be represented in case of such a tie, and curves will be drawn through each value.

  In the figure below, the value of the adjusted FOM for a K-means run decreases steeply until the number of clusters reaches 4, after which it levels out. This suggests that, for this data set, K-means performs optimally for 4 clusters and that any additional clusters produced will not add to the predictive value of the algorithm. FOM is useful in determining the best input parameters for a clustering algorithm.



Method: KMC - Calculated Means
Iterations: 25
Average: true
Time: 17225 ms
Euclidean distance

1--->19.837
2--->6.926
3--->6.983
4--->2.937
5--->2.973
6--->0.968
7--->0.968
8--->0.553
9--->0.288
10--->0.879
11--->2.889
12--->1.842
13--->0.287
14--->0.386
15--->2.916
16--->2.814
17--->0.246
18--->0.233
19--->0.298
20--->0.213
21--->0.289
22--->0.239
23--->0.25
24--->0.262
25--->0.492

Fig. 21. FOM: FOM vs. Number of Clusters graph for the KMC algorithm.

- **Principal Components Analysis** (**Raychaudhuri et al. 2000**) is available from the Analysis menu or the toolbar (**PCAE** for experiments, and **PCAG** for genes). PCA allocates the genes into distinct groups that account for as much of the variance in the gene expression data as possible. Once the calculations are complete, select the PCA node under Analysis to view the PCA results. 3D view is the primary PCA display, and is a three dimensional view. The display can be rotated and shifted by left dragging or right dragging respectively. Right clicking on the 3D view node will display a popup menu that allows the user to change the 3D view's display options and create a selection area (essentially a cube) to define a cluster. PC plots, PC information and Eigenvalues detail the calculations behind the construction of the display.

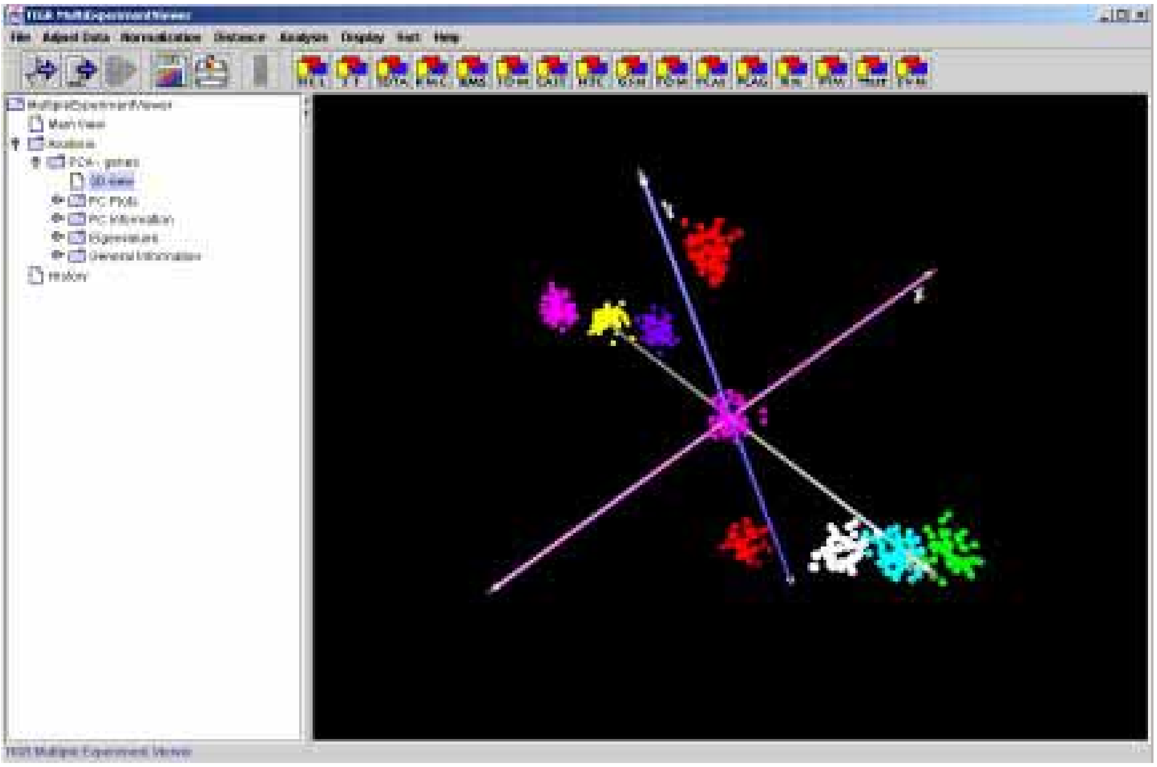


Fig. 22. PCAG: 3D View.

- **Relevance Networks** (**Butte et al. 2000**) are available from the Analysis menu or from the toolbar (**RN**). A relevance network is a group of genes whose expression profiles are highly correlated with one another. Each pair of genes related by a correlation coefficient larger than a minimum threshold and smaller than a maximum threshold (assigned in the module dialog box) is connected by a line. Genes with low-entropy expression profiles (containing the least-uniformly distributed values) can be excluded from the analysis, to minimize the bias in correlation coefficient which can be caused by outlier values. If the "Use Filter" option in the dialog box is selected, genes with entropy values of a higher percentile than that specified by the filter percent parameter will be cut out.

  A graphical representation of the entire network of clusters can be viewed under the Network tab (fig. X). Right-clicking on the graph will bring up a menu containing navigation and selection options. Using the "select" menu option, genes can be selected by gene ID or by the number of links to other genes. Clicking on a node in the graph will bring up the spot information dialog corresponding to that node. The "Expression Images" node on the navigation tree contains sub-nodes, each of which is a cluster of genes consisting of a hub gene linking out to other genes in the network. The number of genes in each cluster is shown in parentheses next to the cluster number. The number of links out from the hub gene is one less than the number of genes in the cluster. This feature enables the user to focus on genes that have a large number of connections to other genes. The "Relevance Subnets" tab on the navigation tree contains sub-nodes that correspond to the subnets of the network, in i.e., groups of genes in which each gene is connected to at least one other gene. Right-clicking on any of these views brings up menus with options for display and selection.

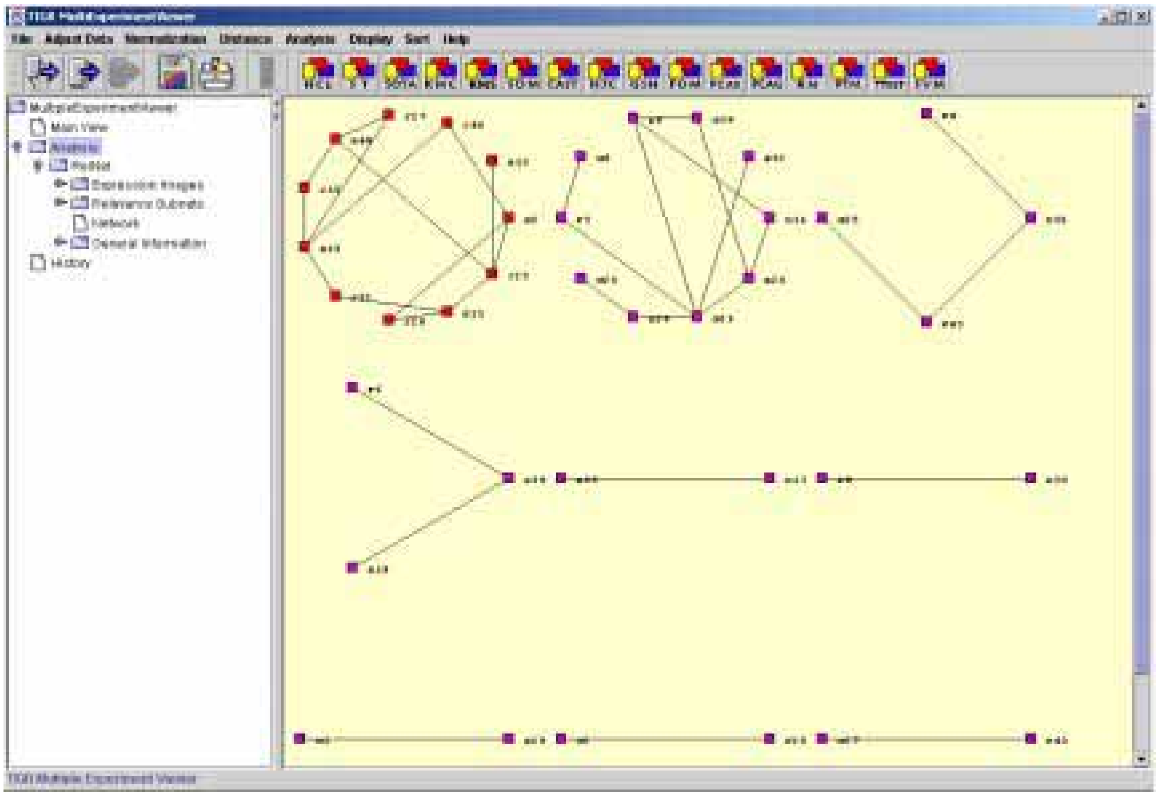
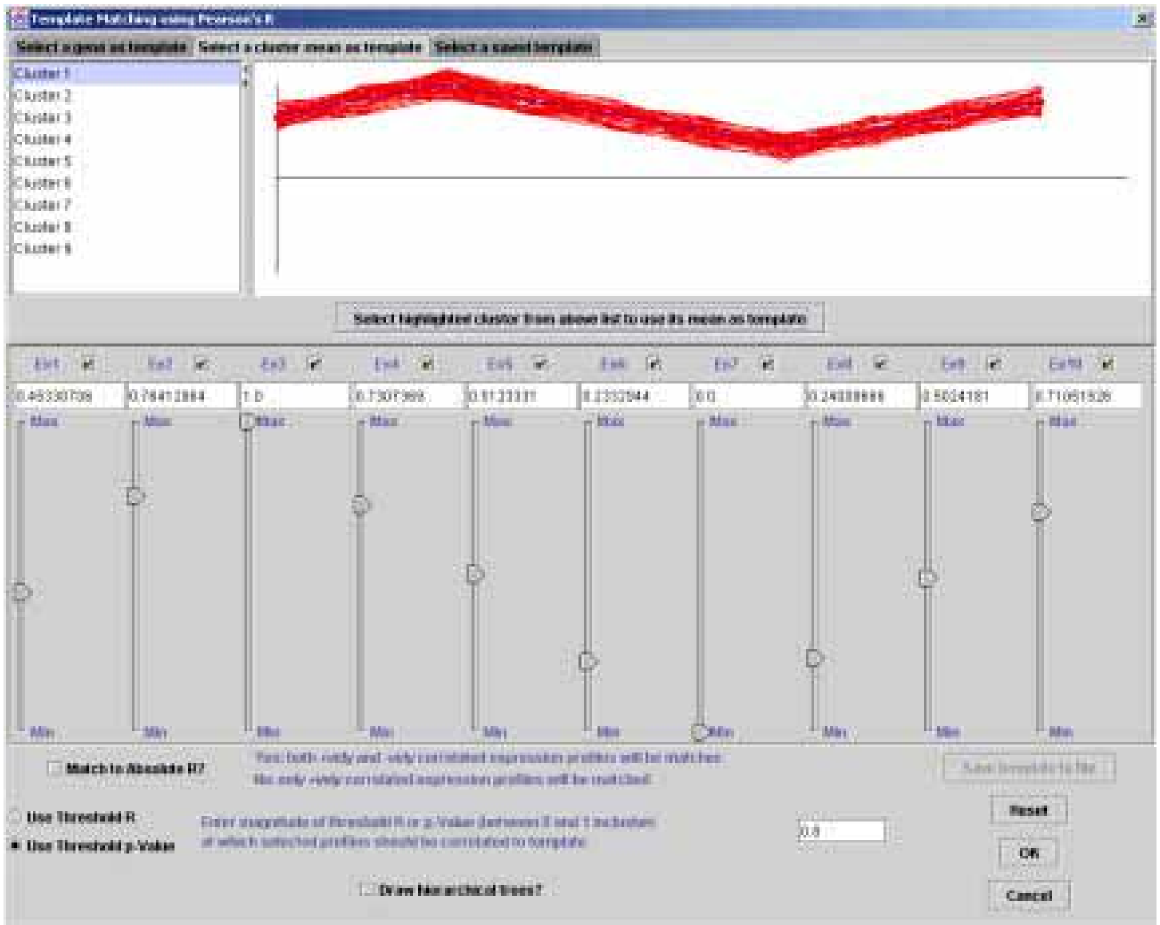


Fig. 23. Relevance Networks: Network View.

- **Template Matching** (**Pavlidis and Noble 2001**) is available from the Analysis Menu (Pavlidis Template Matching) or the toolbar (**PTM**). The user can specify a template expression profile for a gene (a series of relative expression ratios between 0 and 1), and the data set will be searched for matches to the template, based on the Pearson Correlation between the template and the genes in the data set. The template profile can be specified in one of several ways: 1) by selecting one of the genes in the data set as a template from the list on the upper left, and then clicking the "Select highlighted gene" button; 2) doing the same thing with one of the cluster means, assuming that clusters have already been set by some other method; 3) entering values between 0 and 1 in the text input fields above the slider bars corresponding to each experiment; or, 4) Adjusting the slider bars to the desired values. Matches can be made by considering either the signed or the unsigned values of correlation coefficient (using the checkbox labeled "Match to Absolute R?"), and the threshold criterion for matching can be either the magnitude of the correlation coefficient, or the significance (p-value) of the correlation coefficient.

  Template matching is particularly useful when the researcher is searching for a specific expression pattern. Applying this method with the input parameters in the previous figure gives the following output, where the first panel on the right corresponds to genes that matched the template, and the second panel to genes that did not match.


Template Matching using Pearson's R
Select a gene as template
Select a cluster mean as template
Select a saved template
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Select highlighted cluster from above list to use its mean as template
Max
Min
Save template to file
Reset
OK
Cancel

Ex1 ☑ 0.45330708
Ex2 ☑ 0.78412884
Ex3 ☑ 1.0
Ex4 ☑ 0.7307969
Ex5 ☑ 0.6123331
Ex6 ☑ 0.2332944
Ex7 ☑ 0.0
Ex8 ☑ 0.24338888
Ex9 ☑ 0.5024181
Ex10 ☑ 0.71051526

□ Match to Absolute R?

Yes: both +vely and -vely correlated expression profiles will be matched
No: only +vely correlated expression profiles will be matched

○ Use Threshold R
● Use Threshold p-Value

Enter magnitude of threshold R or p-Value (between 0 and 1 inclusive) of which selected profiles should be correlated to template 0.8

□ Draw hierarchical trees?

Fig. 24. Template Matching (PTM) initialization dialog.

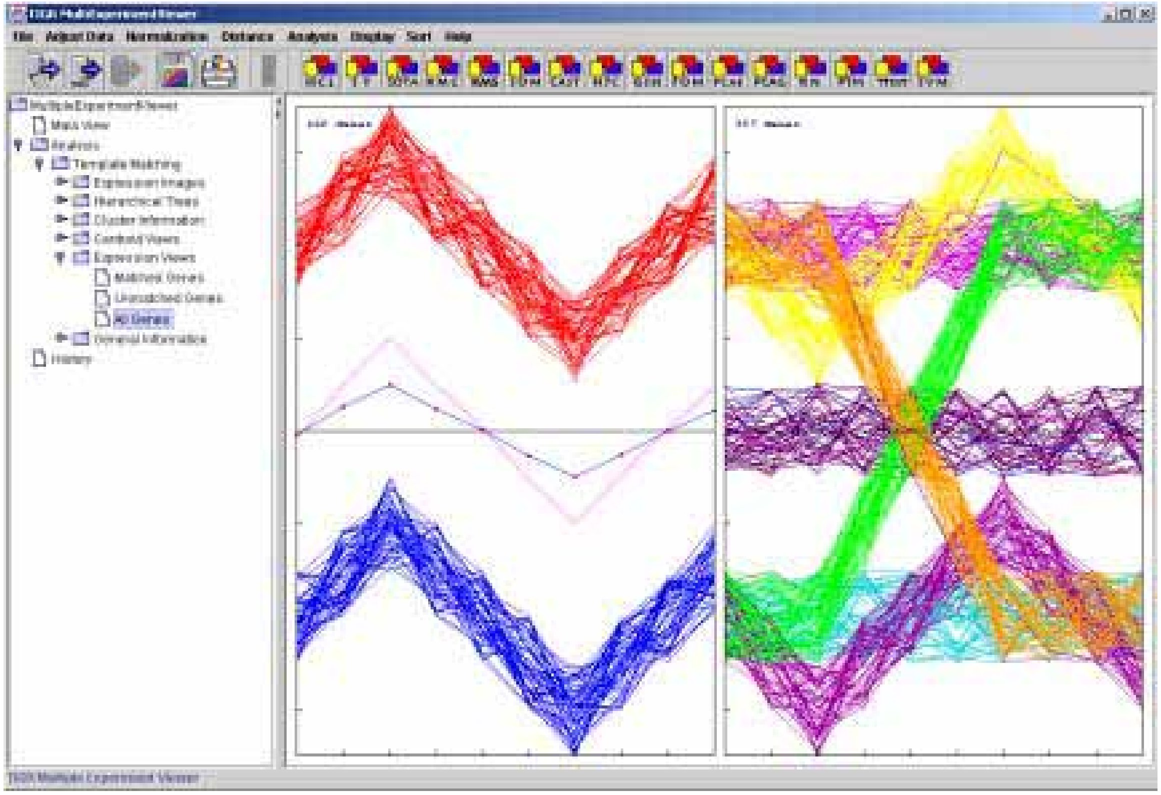
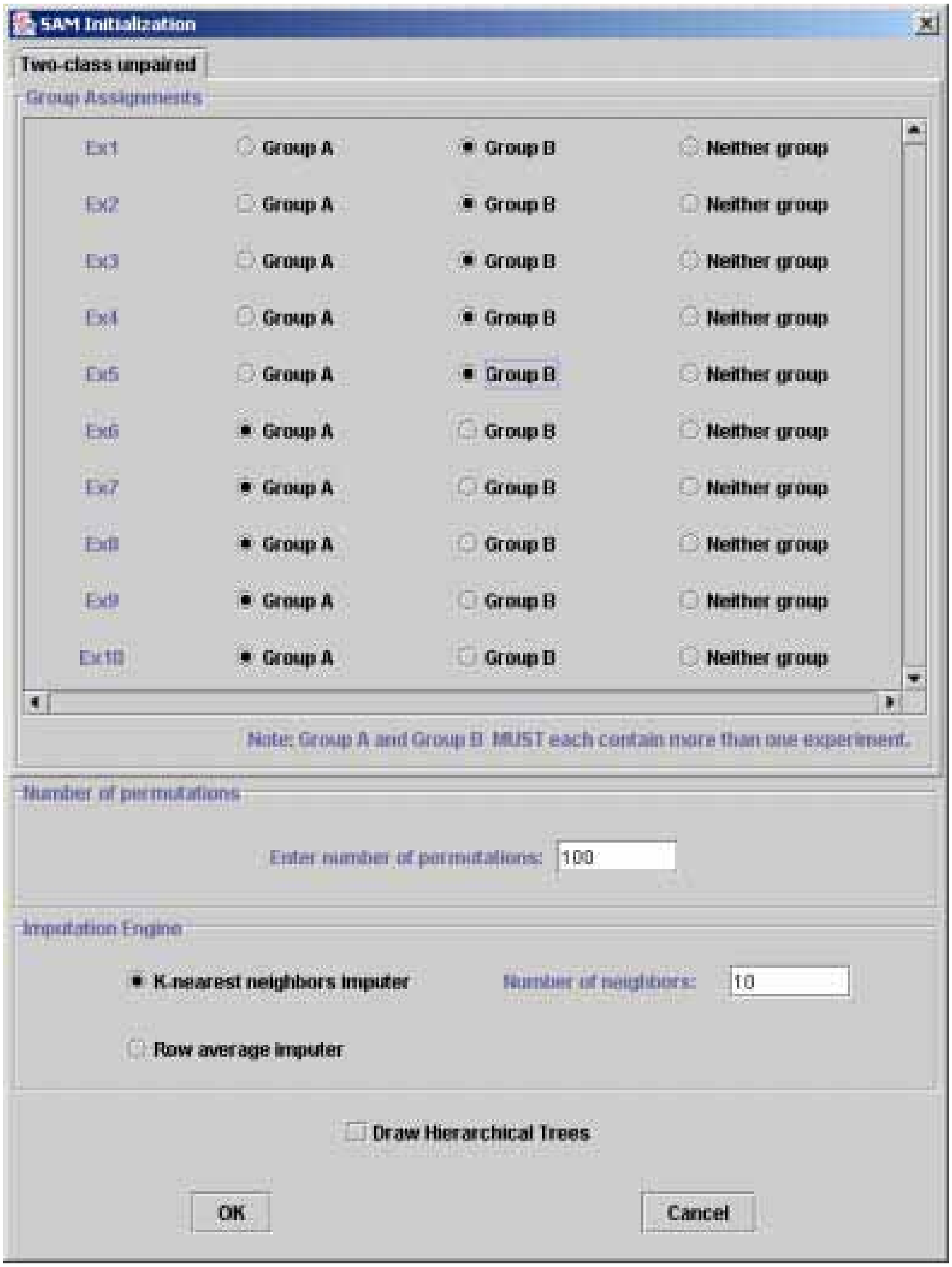


Fig. 25. PTM results: Expression View.

- **Significance Analysis of Microarrays (SAM)** (Tusher et al. 2001)
  SAM can be used to pick out significant genes based on differential expression between sets of experiments. It is useful when there is an *a-priori* hypothesis that some genes will have significantly different mean expression levels between different sets of samples. For example, one could look at differential gene expression between tissue types, or differential response to exposure to a perturbation between groups of test subjects. A valuable feature of SAM is that it gives estimates of the False Discovery Rate (FDR), which is the proportion of genes likely to have been identified by chance as being significant. Furthermore, SAM is a very interactive algorithm. It allows users to eyeball the distribution of the test statistic, and then set thresholds for significance (through the tuning parameter delta) after looking at the distribution. The ability to dynamically alter the input parameters based on immediate visual feedback, even before completing the analysis, should make the data-mining process more sensitive.

  Currently, SAM is implemented for the two-class unpaired design, where experiments fall in one of two groups, and the subjects are different between the two groups (analogous to a between subjects t-test). The initialization dialog box **(Fig. 26**) is similar to the t-test dialog.



SAM Initialization

Two-class unpaired

Group Assignments

Ex1	○ Group A	● Group B	○ Neither group
Ex2	○ Group A	● Group B	○ Neither group
Ex3	○ Group A	● Group B	○ Neither group
Ex4	○ Group A	● Group B	○ Neither group
Ex5	○ Group A	● Group B	○ Neither group
Ex6	● Group A	○ Group B	○ Neither group
Ex7	● Group A	○ Group B	○ Neither group
Ex8	● Group A	○ Group B	○ Neither group
Ex9	● Group A	○ Group B	○ Neither group
Ex10	● Group A	○ Group B	○ Neither group

Note: Group A and Group B MUST each contain more than one experiment.

Number of permutations

Enter number of permutations: 100

Imputation Engine

● K-nearest neighbors imputer Number of neighbors: 10

○ Row average imputer

□ Draw Hierarchical Trees

Fig. 26. SAM Initialization Dialog.

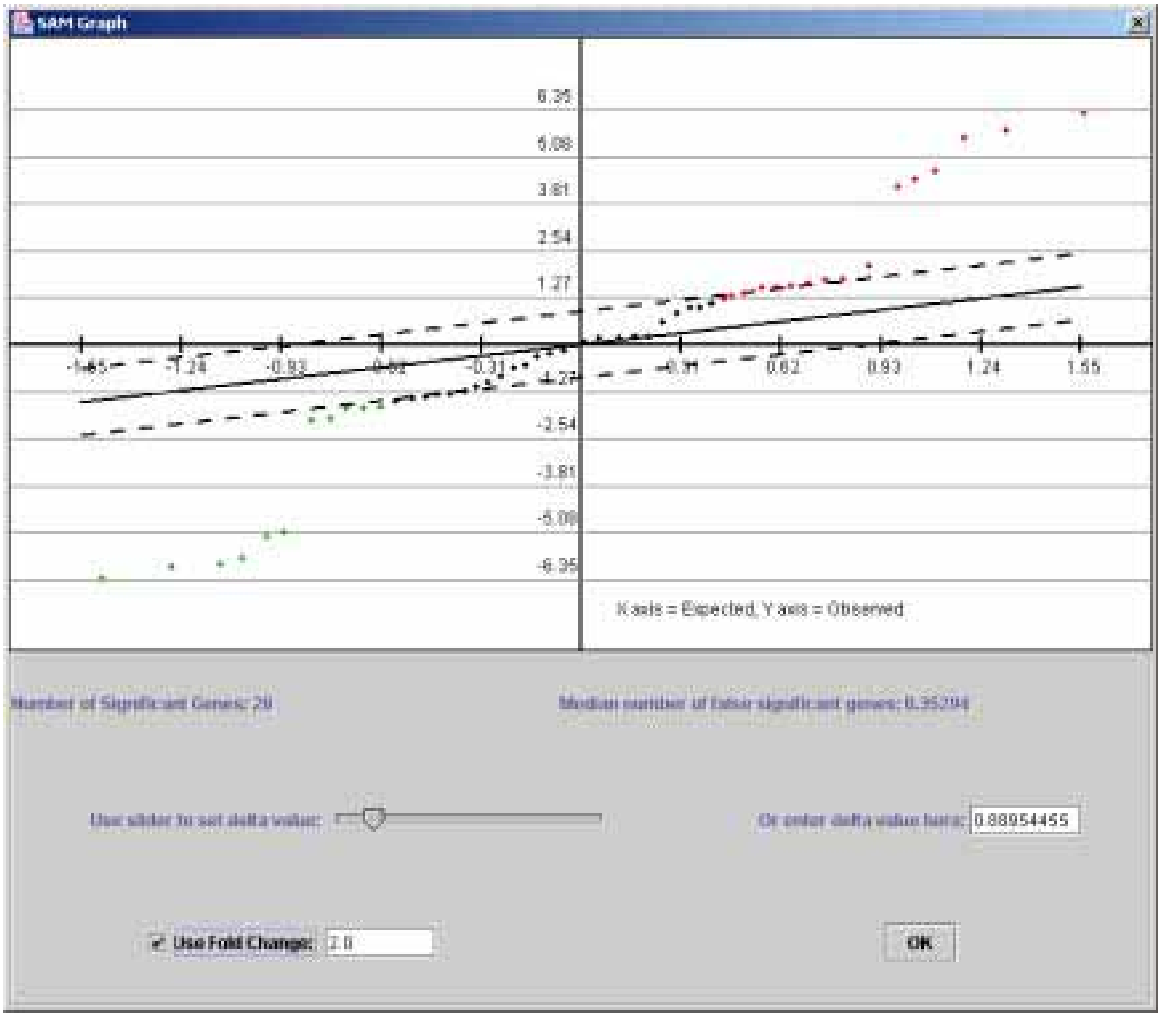
The user inputs the group memberships of the experiments in the top panel. In the two-class design, genes will be considered to be “positive significant” if their mean expression in group B is significantly higher than in group A. They will be considered “negative significant” if the mean of group A significantly exceeds that of group B.

The data for each gene are permuted, and a test statistic *d* is computed for both the original and the permuted data for each gene. In the two-class unpaired design, *d* is analogous to the t-statistic in a t-test, in that it captures the difference among mean expression levels of experimental conditions, scaled by a measure of variance in the data. Missing values in the input data matrix are imputed by one of two methods: 1) **Row average**: replacing missing expression measurements with the mean expression of a row (gene) across all columns (experiments), OR 2) **K-nearest neighbors**: where the "K" most similar genes (using Euclidean distance) to the gene with a missing value are used to impute the missing value.

SAM generates an interactive plot (**Fig. 27**) of the observed vs. expected (based on the permuted data) *d*-values. The user can change the value of the tuning parameter delta using either the slider bar or the text input field below the plot. *Delta* is a vertical distance (in graph units) from the solid line of slope 1 (i.e., where observed = expected). The two dotted lines represent the region within +/- delta units from the "observed = expected" line. The genes whose plot values are represented by black dots are considered non-significant, those colored red are positive significant, and the green ones are negative significant. The user can also choose to apply a **fold change** criterion. In this case, in addition to satisfying the delta criterion, a gene will also have to satisfy the following condition to be considered significant:

For a given fold change **F**,

[Mean (unlogged group B values) / Mean (unlogged group A values)] ≥ F (for positive significant genes), or ≤ 1/F (for negative significant genes).



Number of Significant Genes: 20

Median number of false significant genes: 0.35294

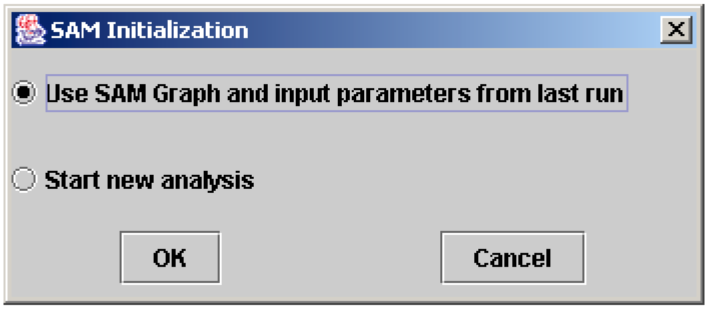
Use slider to set delta value:

Or enter delta value here: 0.88954455

☑ Use Fold Change: 2.0

Fig. 27. SAM Graph

If SAM has been used at least once during an MeV session, the input parameters used by that run of SAM and the resulting graph can be called up by default from the dialog box shown in **Fig. 28**, thus bypassing the need to run SAM again for that set of parameters. This feature is useful because the computations leading up to the graph can be time-consuming, and need not be repeated for a given set of input parameters.



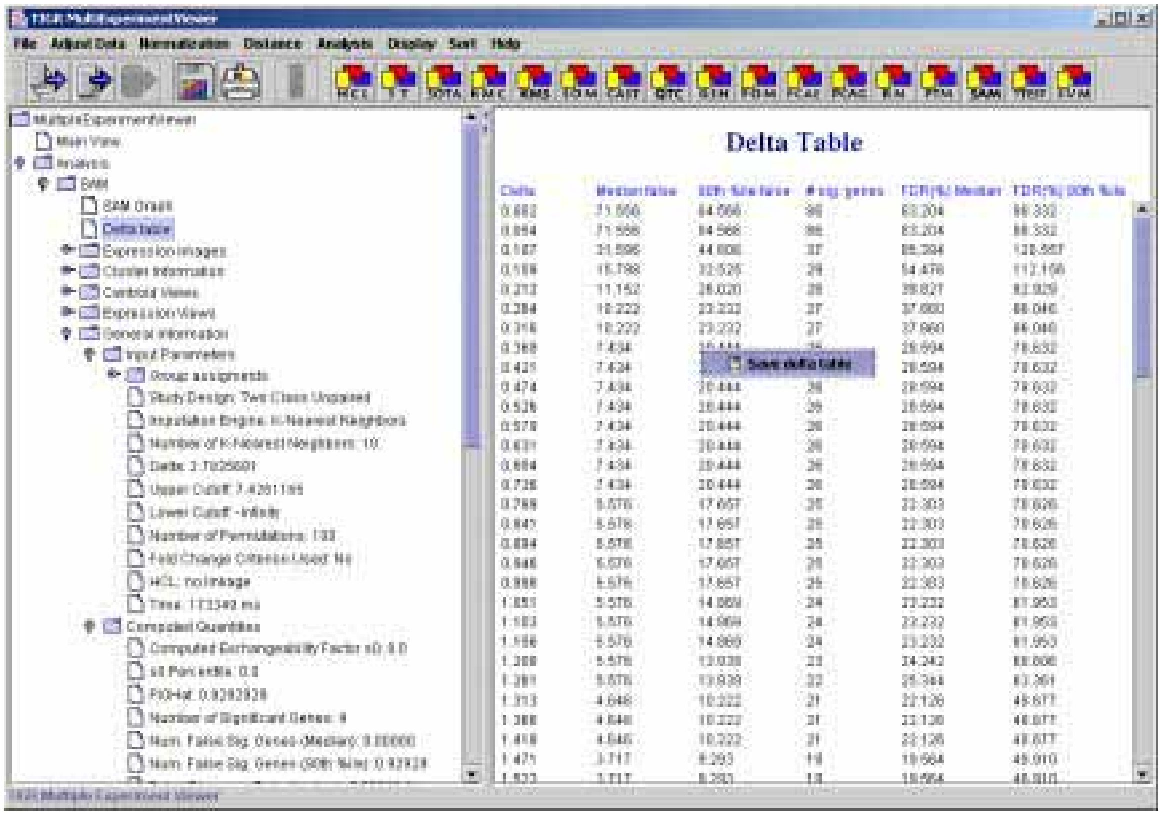
SAM Initialization

● Use SAM Graph and input parameters from last run

○ Start new analysis

Fig. 28. SAM Parameter Recall Dialog

In addition to the standard viewers and information tabs, SAM also outputs a SAM graph viewer, as well as a Delta table viewer (**Fig 29**), which contains output information for a range of delta values. This information can be saved as a tab-delimited text file by right-clicking on the table. The clusters saved from the other SAM viewers will store gene-specific SAM statistics in addition to the annotation and expression measurements stored in clusters from most other modules.



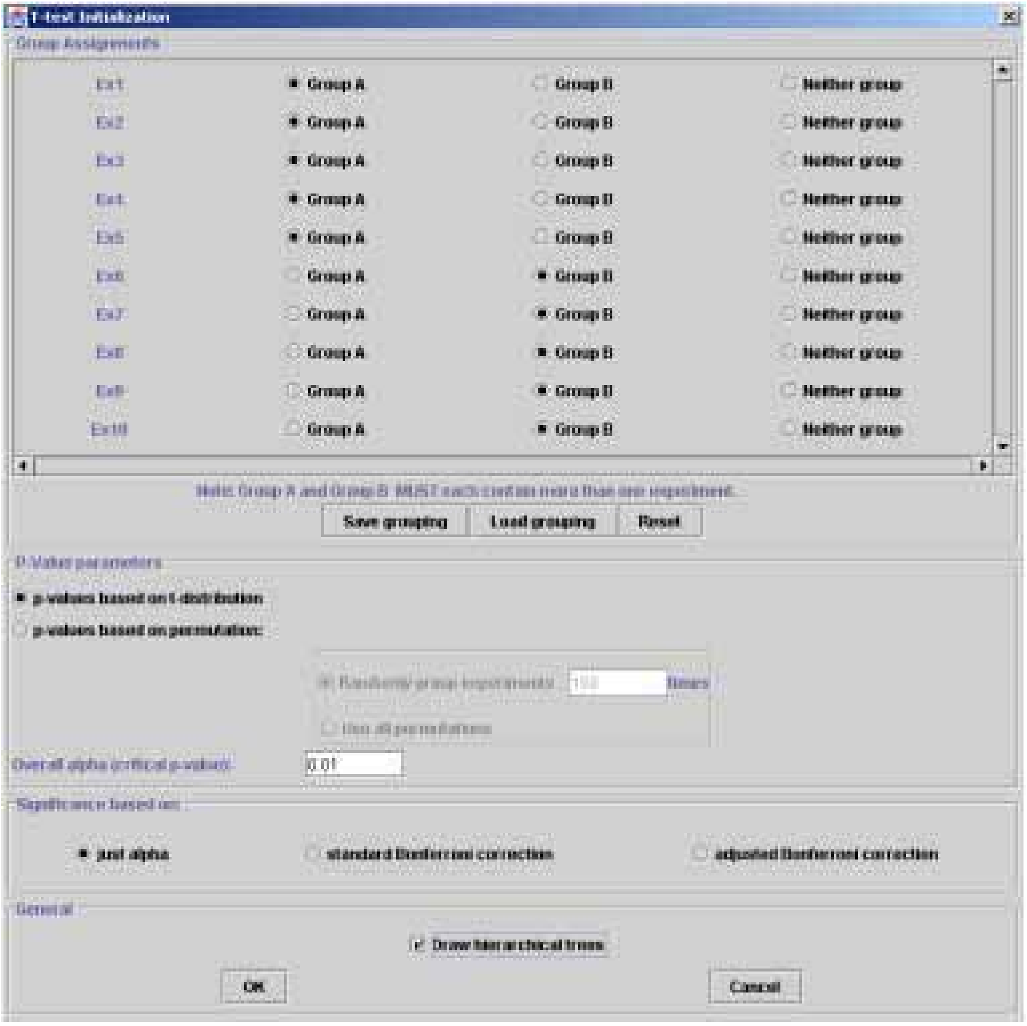
MultipleExperimentViewer
Main View
Analysis
SAM
SAM Graph
Delta table
Expression Images
Cluster Information
Centroid Views
Expression Views
General Information
Input Parameters
Group assignments
Study Design: Two Class Unpaired
Imputation Engine: K-Nearest Neighbors
Number of K-Nearest Neighbors: 10
Delta: 3.7025601
Upper Cutoff: 7.4261195
Lower Cutoff: -Infinity
Number of Permutations: 100
Fold Change Criterion Used: No
HCL: no image
Time: 173349 ms
Computed Quantities
Computed Exchangeability Factor s0: 0.0
s0 Percentile: 0.0
Pi0Hat: 0.9292929
Number of Significant Genes: 9
Num. False Sig. Genes (Median): 0.00000
Num. False Sig. Genes (90th %ile): 0.92929

Delta Table

Delta	Median false	90th %ile false	# sig. genes	FDR(%) Median	FDR(%) 90th %ile
0.002	71.556	84.566	86	83.204	98.332
0.054	71.556	84.566	86	83.204	98.332
0.107	21.596	44.606	37	58.394	120.557
0.159	15.798	32.526	29	54.476	112.156
0.212	11.152	26.020	28	39.827	92.929
0.264	10.222	23.232	27	37.860	86.046
0.316	10.222	23.232	27	37.860	86.046
0.368	7.434	20.444	26	28.594	78.632
0.421	7.434			28.594	78.632
0.474	7.434	20.444	26	28.594	78.632
0.526	7.434	20.444	26	28.594	78.632
0.578	7.434	20.444	26	28.594	78.632
0.631	7.434	20.444	26	28.594	78.632
0.684	7.434	20.444	26	28.594	78.632
0.736	7.434	20.444	26	28.594	78.632
0.789	5.576	17.657	25	22.303	70.626
0.841	5.576	17.657	25	22.303	70.626
0.894	5.576	17.657	25	22.303	70.626
0.946	5.576	17.657	25	22.303	70.626
0.998	5.576	17.657	25	22.303	70.626
1.051	5.576	14.869	24	23.232	61.953
1.103	5.576	14.869	24	23.232	61.953
1.156	5.576	14.869	24	23.232	61.953
1.208	5.576	13.939	23	24.242	60.606
1.261	5.576	13.939	22	25.344	63.361
1.313	4.646	10.222	21	22.126	48.677
1.366	4.646	10.222	21	22.126	48.677
1.418	4.646	10.222	21	22.126	48.677
1.471	3.717	8.293	19	19.564	48.910
1.523	3.717	8.293	19	19.564	48.910

Fig. 29. SAM Delta Table Viewer.

- **T-tests** (**Dudoit et al. 2000, Pan 2002**) are available from the Analysis menu or the toolbar (**TTEST**). Experiments can be assigned to one of two groups, and genes that have significantly different mean expression levels between the two groups are assigned to one cluster, while the genes that are not significantly different between the two groups are assigned to another cluster. The user may choose to exclude some experiments from the analysis, which can be done by selecting the "neither group" option for those experiments in the initialization dialog (see screenshot below). T-values are calculated for each gene, and p-values are computed either from the theoretical t-distribution, or from permutations of the data for each gene between the two groups. Whether a gene's mean expression level is significantly different between the two groups is determined either by directly comparing the gene's p-value with the user-specified critical p-value or alpha, or by adjusting for error rate using a standard or adjusted Bonferroni correction (see screenshot below).



T-test Initialization

Group Assignments

Ex1	● Group A	○ Group B	○ Neither group
Ex2	● Group A	○ Group B	○ Neither group
Ex3	● Group A	○ Group B	○ Neither group
Ex4	● Group A	○ Group B	○ Neither group
Ex5	● Group A	○ Group B	○ Neither group
Ex6	○ Group A	● Group B	○ Neither group
Ex7	○ Group A	● Group B	○ Neither group
Ex8	○ Group A	● Group B	○ Neither group
Ex9	○ Group A	● Group B	○ Neither group
Ex10	○ Group A	● Group B	○ Neither group

Note: Group A and Group B MUST each contain more than one experiment.

P-Value parameters

● p-values based on t-distribution

○ p-values based on permutation:

● Randomly group experiments 100 times

○ Use all permutations

Overall alpha (critical p-value): 0.01

Significance based on:

● just alpha ○ standard Bonferroni correction ○ adjusted Bonferroni correction

General

☑ Draw hierarchical trees

Fig. 30. TTEST Initialization Dialog Box.

In the standard Bonferroni correction, the user-specified alpha is divided by the number of genes to give the critical p-value. In the adjusted Bonferroni correction, the t-values for all the genes are ranked in descending order. For the gene with the highest t-value, the critical p-value becomes (alpha / n), where n is the total number of genes; for the gene with the second-highest t-value, the critical p-value will be (alpha/ n-1), and so on. Bonferroni corrections reduce the probability that a non-significant gene will be erroneously picked as significant. This can be a serious issue when many tests are done (which is usually the case in microarray analyses, as there are as many tests as there are genes in the analysis). The standard Bonferroni correction is very stringent and may exclude many genes that are really significant, whereas the adjusted Bonferroni correction is less conservative, and more likely to include significant genes while still controlling the error rate.

Sample output from this module is shown below:

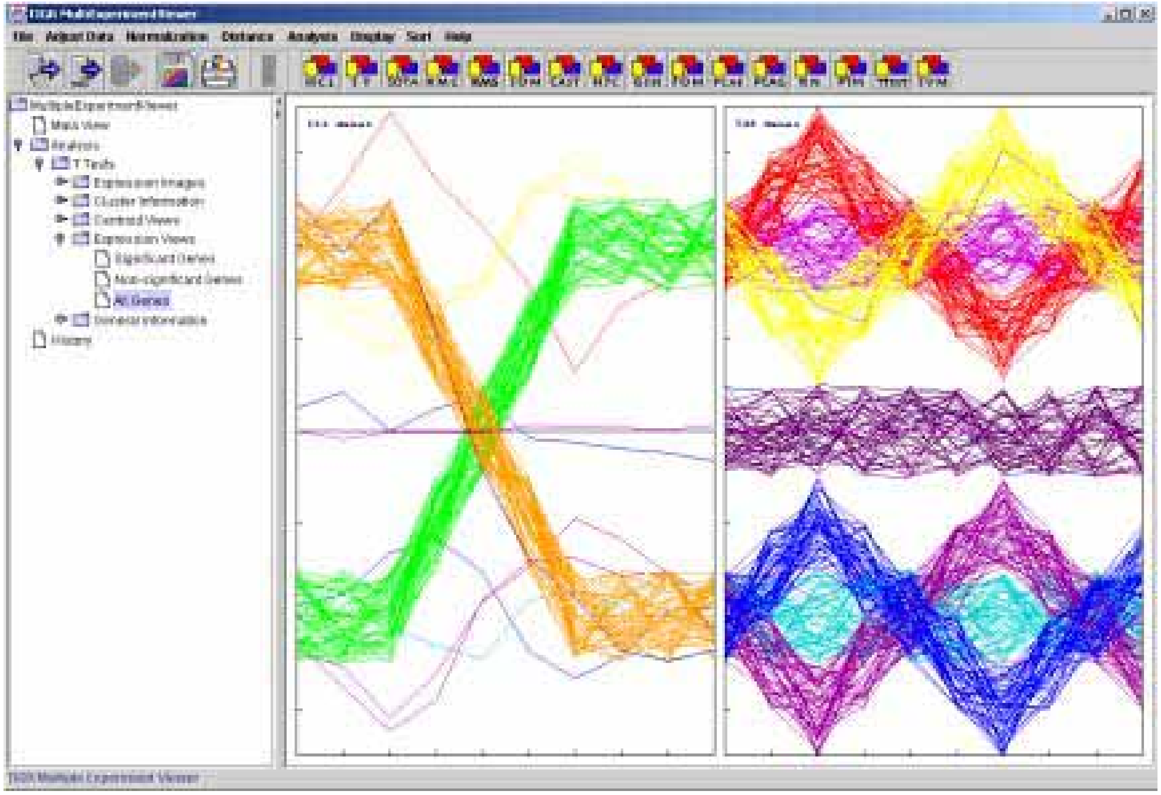


Fig. 31. TTEST Results: Expression View.

- **Support Vector Machines (Brown et al., 2000)**
  Support Vector Machines (**SVM**) is available from the Analysis toolbar. Although SVMs have been used in various fields of study, the use of SVMs for gene expression analysis was described in detail by Brown et al.. SVM is a supervised learning classification technique. The algorithm uses supplied information about existing relationships between members of a subset of the elements to be classified. The supplied information, an initial presumed relationship between a set of elements, coupled with the expression pattern data leads to a binary classification of each element. This presumptive initial classification is defined by the researcher using available information about the elements to be classified such as functional relationships. Following the analysis, each element is considered either in or out of the initial presumptive classification.

The algorithm proceeds through two main phases. The first of these phases, **training**, uses the presumptive classification (supplied knowledge) and the expression data as inputs to produce a set of weights which will be used during the next phase. The second phase, **classification**, uses the weights created during training and the expression data to assign a discriminant or score to each element. Based on this score each element is placed into or out of the class. (fig. 32) This final classification of elements will either support or oppose the initial classification. After classification, inclusion into the presumed classification reveals that the element had an expression profile which had sufficient similarity to the dominant profile of the elements which were presumed to be related to be considered as part of that classification. Note that this partition will depend on algorithm parameters which can roughly determine the stringency of the classification process.

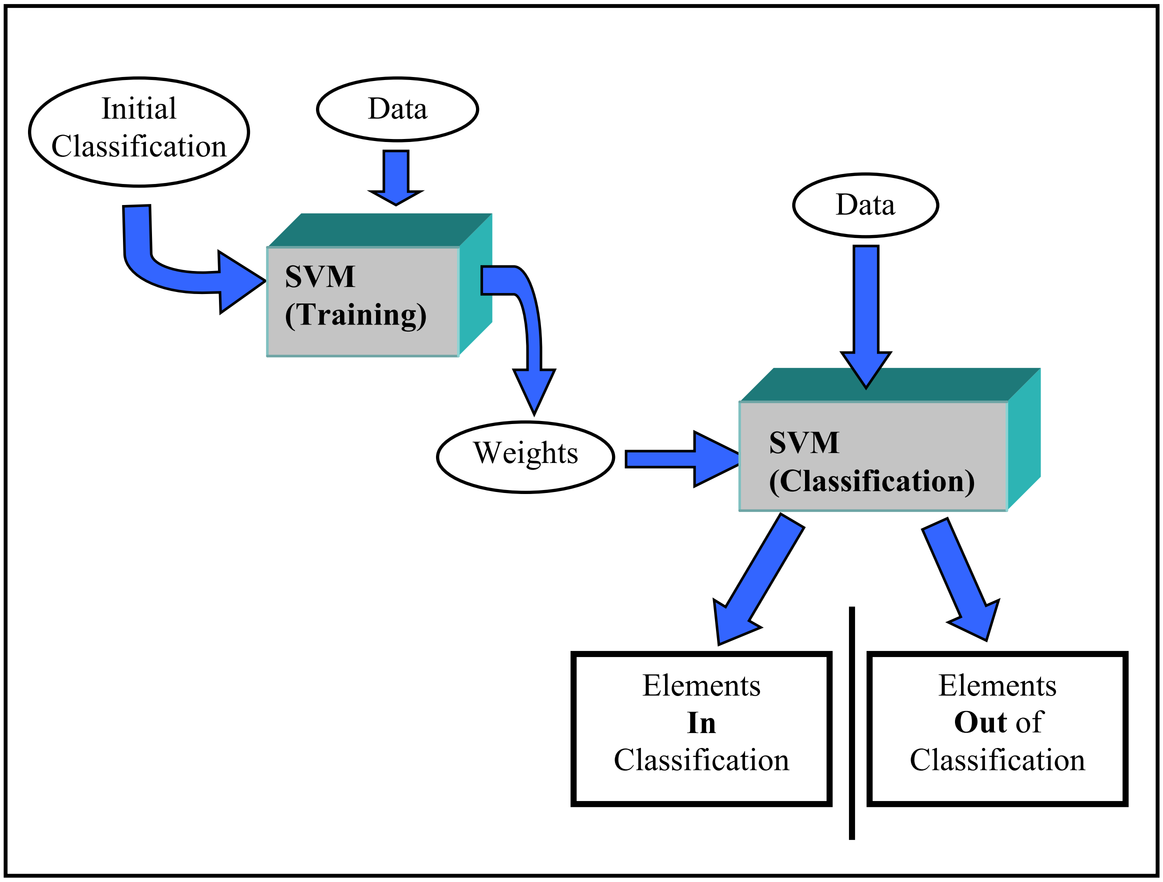
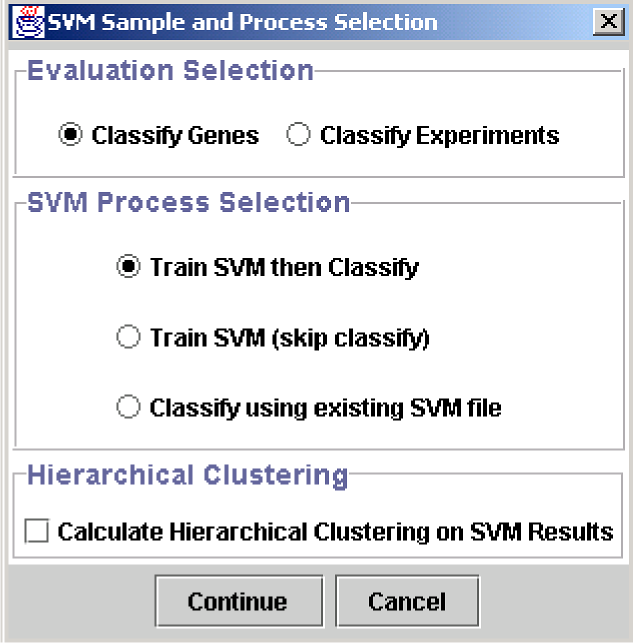


Fig. 32. SVM Process Overview

SVM Dialog Overview

The initial dialog, *SVM Process Selection Dialog* (fig. 33), can be used to select to classify genes or experiments and to perform one or both phases of the algorithm. The Train and Classify option allows one to run both phases of the algorithm. Starting with a presumptive classification and expression data the result is a final classification of each element. The Train only option produces a list of weights which can be stored as an 'SVM' file along with training parameters so that they can be applied to data to classify at a later time. The Classify only option prompts the user for an SVM file of weights and parameters and results in final classification. The user also has an option to produce hierarchical trees the two groups of elements resulting from classification. One group comprised of elements determined to be related to the elements in the initial classification and the other group comprised of elements that have expression data that does not support a such a relationship.

The second dialog, *SVM Initalization*, (fig. 34) is used during either the Train and Classify mode or the Train Only mode. The upper portion is used to indicate whether the initial presumptive classification will be defined using the SVMClassification Editor or supplied as an *SVC file*.



SVM Sample and Process Selection

Evaluation Selection

● Classify Genes ○ Classify Experiments

SVM Process Selection

● Train SVM then Classify

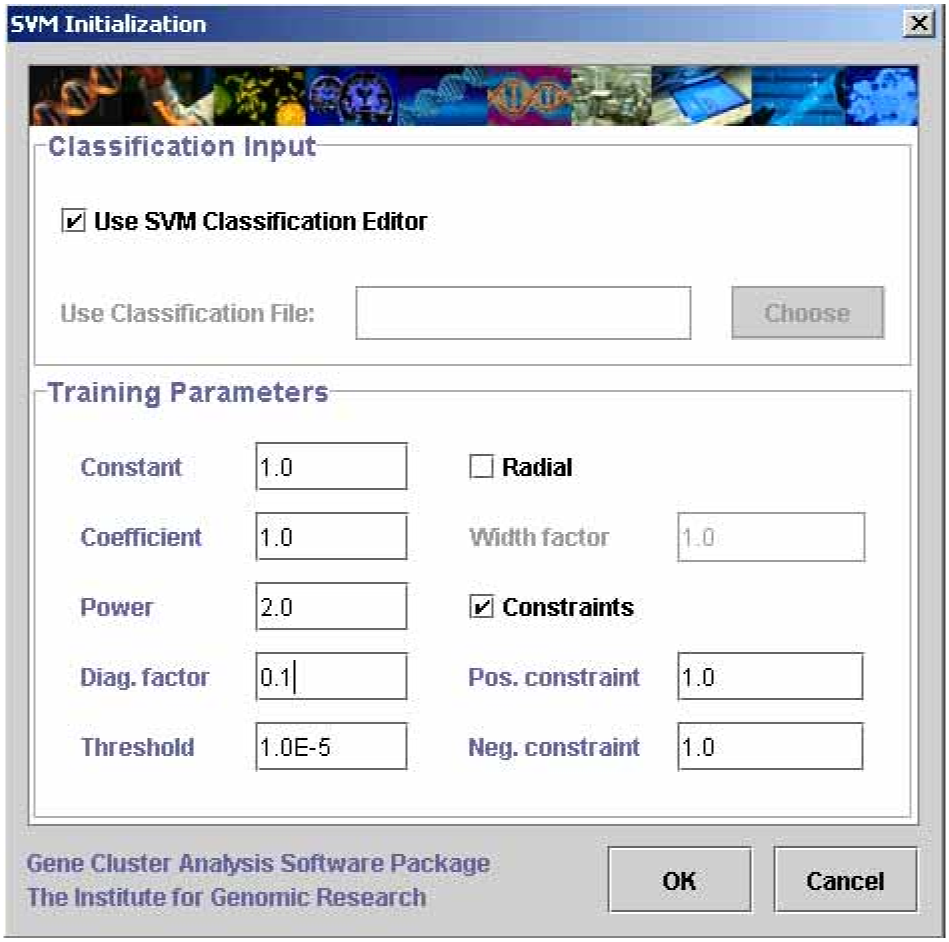
○ Train SVM (skip classify)

○ Classify using existing SVM file

Hierarchical Clustering

□ Calculate Hierarchical Clustering on SVM Results

Fig. 33. SVM Process Selection Dialog



SVM Initialization

Classification Input

☑ Use SVM Classification Editor

Use Classification File:

Training Parameters

Constant: 1.0

Coefficient: 1.0

Power: 2.0

Diag. factor: 0.1

Threshold: 1.0E-5

□ Radial

Width factor: 1.0

☑ Constraints

Pos. constraint: 1.0

Neg. constraint: 1.0

Gene Cluster Analysis Software Package
The Institute for Genomic Research

Figure 34. SVM Initialization Dialog

The *SVC file* format is a tab delimited text file with the following columns for each element,

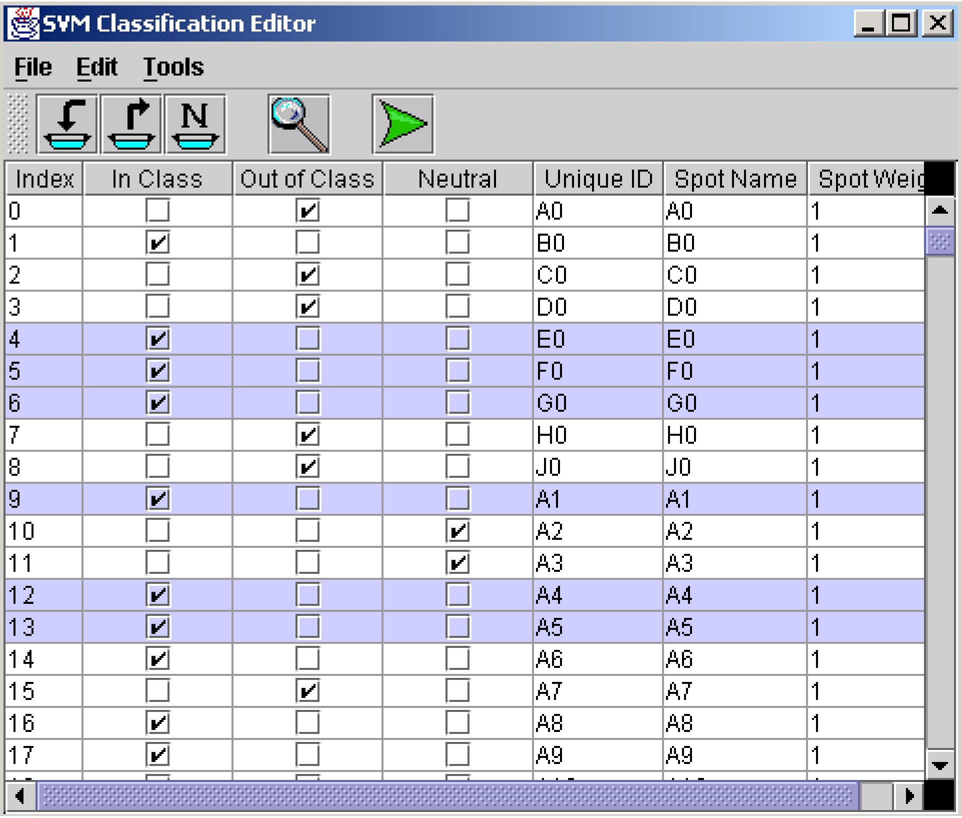
1.) Index - a sequential integer index.

2.) Classification - an integer value indicating class membership.

(1 = in initial classification, 0 = neutral, -1 = out of initial classification)

3.) Optional annotation columns.

The SVMClassification Editor (fig. 35) allows one to use searches on supplied annotation as well as SVC files to assign membership to the initial presumptive classification. Elements not strictly in the initial classification can be labeled as either not in the initial class or as neutral in the case where not information about the nature of an element exists. The editor allows the user to sort the element list based on classification or annotation fields. The constructed initial classification can be stored in SVC format and later reloaded to allow alterations to produce what could be several initial classifications for a given study. The SVC files, once created, can be used to supply the initial classification thereby skipping the editor step. If the editor is used a button or menu selection launches the algorithm based on the current classification selection.



Index	In Class	Out of Class	Neutral	Unique ID	Spot Name	Spot Weig
0	□	☑	□	A0	A0	1
1	☑	□	□	B0	B0	1
2	□	☑	□	C0	C0	1
3	□	☑	□	D0	D0	1
4	☑	□	□	E0	E0	1
5	☑	□	□	F0	F0	1
6	☑	□	□	G0	G0	1
7	□	☑	□	H0	H0	1
8	□	☑	□	J0	J0	1
9	☑	□	□	A1	A1	1
10	□	□	☑	A2	A2	1
11	□	□	☑	A3	A3	1
12	☑	□	□	A4	A4	1
13	☑	□	□	A5	A5	1
14	☑	□	□	A6	A6	1
15	□	☑	□	A7	A7	1
16	☑	□	□	A8	A8	1
17	☑	□	□	A9	A9	1

Fig. 35. SVM Classification Editor

The *SVM Initialization Dialog* (fig. 34), is use to define parameters used for creating the kernel matrix. The following is an overview of the training parameters.

Kernel Construction Parameters

Parameters for Polynomial Kernel [K(x,y) = b*(s(x,y) + a)**c]
Constant (a) – additive constant
Coefficient (b) – multiplicative coefficient
Power (c) – kernal function power

Parameters for Radial Kernel [$K(x,y) = \exp(- (\|x – y\|^2)/(2w^2))$]
Radial – selects to use a radial basis kernal
Width (w) – radial width factor

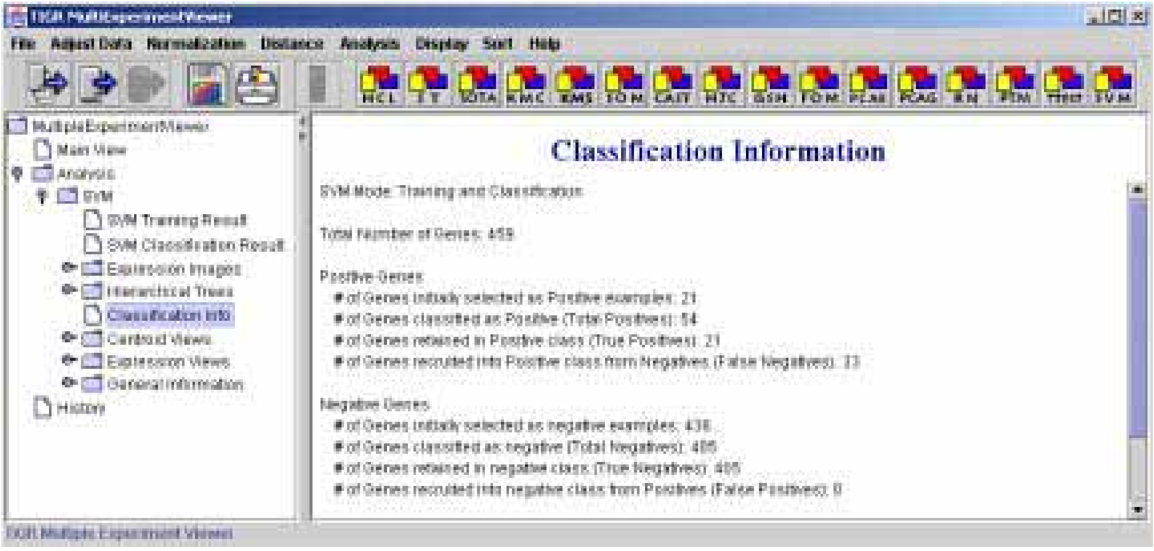
Training Parameters

Diagonal Factor – Constant added to main diagonal
Threshold – stopping criteria for weight optimization phase of training
Constraints – selects to apply limits to weights

- **Positive Constraint** - upper limit to produced weights
- **Negative Constraint** - lower limit to produced weights

SVM Output

The final result of an SVM run depends upon the process run. Training results in a set of weights that can be viewed along with the parameters for kernel construction. Note that from this viewer the training results can be saved as an *SVM file*. Classification results in a viewer that indicates each element's discriminant value and a final classification. The *SVM Classification Information Viewer* describes how many elements were initially selected as positive examples and how many elements were later recruited into the positive and negative classifications as well as other overview statistics.



Classification Information

SVM Mode: Training and Classification

Total Number of Genes: 459

Positive Genes
  # of Genes initially selected as Positive examples: 21
  # of Genes classified as Positive (Total Positives): 54
  # of Genes retained in Positive class (True Positives): 21
  # of Genes recruited into Positive class from Negatives (False Negatives): 33

Negative Genes
  # of Genes initially selected as negative examples: 438
  # of Genes classified as negative (Total Negatives): 405
  # of Genes retained in negative class (True Negatives): 405
  # of Genes recruited into negative class from Positives (False Positives): 0

Figure 36. Classification Information Viewer

Expression image viewers reveal which elements have been recruited into each of the final classification sets by coloring the annotation red. Other result viewers are essentially the same as those described in the K-Means clustering section.

REFERENCES

Ben-Dor, A., R. Shamir, and Z. Yakhini 1999. Clustering gene expression patterns. Journal of Computational Biology 6:281-297.

Brown, M.P., W.N. Grundy, D. Lin, N. Cristianini, C.W. Sugnet, T.S. Furey, M. Ares, Jr., and D. Haussler 2000. Knowledge-based analysis of microarray gene expression data by using support vector machines. Proceedings of the National Academy of Sciences USA 97: 262-267.

Butte, A.J., P. Tamayo, D. Slonim, T.R.Golub, I.S. Kohane 2000. Discovering functional relationships between RNA expression and chemotherapeutic susceptibility using relevance networks. Proceedings of the National Academy of Sciences USA 97:12182–12186.

Dopazo J., J. M. Carazo 1997. Phylogenetic reconstruction using an unsupervised growing neural network that adopts the topology of a phylogenetic tree. J. Mol. Evol. 44:226-233.

Dudoit, S., Y.H. Yang, M.J. Callow, and T. Speed 2000. Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. Technical Report 2000, Statistics Dept., Univ. of California, Berkeley.

Eisen, M.B., P.T. Spellman, P.O. Brown, and D. Botstein 1998. Cluster analysis and display of genome-wide expression patterns. Proceedings of the National Academy of Sciences USA 95:14863-14868.

Hastie,T., R. Tibshirani, M.B. Eisen, A. Alizadeh, R. Levy, L. Staudt, W.C. Chan, D. Botstein, P. Brown 2000. 'Gene shaving' as a method for identifying distinct sets of genes with similar expression patterns. Genome Biology 1:RESEARCH0003.

Herrero, J., A. Valencia, and J. Dopazo 2001. A hierarchical unsupervised growing neural network for clustering gene expression patterns. Bioinformatics 17(2):126-136

Heyer, L.J., S. Kruglyak, and S. Yooseph 1999. Exploring expression data: identification and analysis of co expressed genes. Genome Research 9:1106-1115.

Pan, W. 2002. A comparative review of statistical methods for discovering differentially expressed genes in replicated microarray experiments. Bioinformatics 18: 546-554.

Pavlidis, P., and W.S. Noble 2001. Analysis of strain and regional variation in gene expression in mouse brain. Genome Biology 2:research0042.1-0042.15

Raychaudhuri, S., J. M. Stuart, & R. B. Altman 2000. Principal components analysis to summarize microarray experiments: application to sporulation time series. Pacific Symposium on Biocomputing 2000, Honolulu, Hawaii, 452-463.
Available at http://smi-web.stanford.edu/pubs/SMI_Abstracts/SMI-1999-0804.html

Soukas, A., P. Cohen, N.D. Socci, and J.M. Friedman 2000. Leptin-specific patterns of gene expression in white adipose tissue. Genes and Development 14:963-980.

Tamayo, P., D. Slonim, J. Masirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E.S. Lander, and T.R. Golub 1999. Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. Proceedings of the National Academy of Sciences USA 96:2907-2912.

Tusher, V.G., R. Tibshirani and G. Chu. 2001. Significance analysis of microarrays applied to the ionizing radiation response. Proceedings of the National Academy of Sciences USA 98: 5116-5121.

Yeung, K.Y., D.R. Haynor, and W.L. Ruzzo 2001. Validating clustering for gene expression data. Bioinformatics 17:309-318.