J-Express pro 2.8

User manual

Version 1.1

User's Manual

Table of contents

1 Introduction

J-Express Pro in use.

The J-Express Pro package allows the user to load a data set resulting from a set of microarray experiments and to apply a number of analysis methods, view the results, and produce publication quality figures. The analysis methods include clustering methods (hierarchical and K-means clustering), projection methods (Principal Component Analysis), correspondence analysis, and self-organizing maps. J-Express Pro also provides feature selection methods to identify genes differentiating between classes of arrays. A scripting interface is also available, allowing streamlining and automatically repeating standard analyses. J-Express supports import of MAGE-ML data, facilitating exchange of data with microarray databases including ArrayExpress and BASE. J-Express Pro has an integral project management functionality that helps the user keep track of the datasets, analyses performed, etc. A Server/client system built into J-express allows multiple users to work on a single project simultaneously.

2 Getting Started

2.1 Downloading and installing J-Express Pro

2.1.1 System requirements

J-Express Pro is developed in JAVA ™, and will run on any system that supports the JAVA ™ Virtual Machine version 1.4 or above. These include: Microsoft Windows 98/ME/NT, Sun Solaris, Red Hat Linux, and others. J-Express Pro requires about 50 MB of hard disk space for installation (if a JAVA ™ Virtual Machine is already installed). Suggested minimum requirements (for PC systems):

- *Cpu ~1.5 Ghz*

- *2.0 GB RAM*

Note: Larger datasets can have higher memory requirements.

2.1.2 Download and setup

J-Express Pro is available for download at the web page http://www.molmine.com > download. Follow the link that matches your system. If you need to install the JAVA ™ Virtual Machine, select the appropriate link. If your operating system is not listed, choose the pure JAVA ™ installation file. Note: installation of the Virtual Machine on the Solaris and Linux platforms may require administrator (root) privileges. Please contact your local system administrator if necessary. Follow the instructions on the download page for your platform to complete the download and start the install application.

Initial installation screen

The installation application automatically extracts the files it needs. After this process is complete the program displays the window above. You can cancel the install process at any time before completion by clicking cancel. If you cancel the installation process you will need to restart the install application if you wish to install J-Express Pro at a later date. If you want to return to a previous screen in the installer click **<<Back**. To proceed with the installation, click **Next>>**.
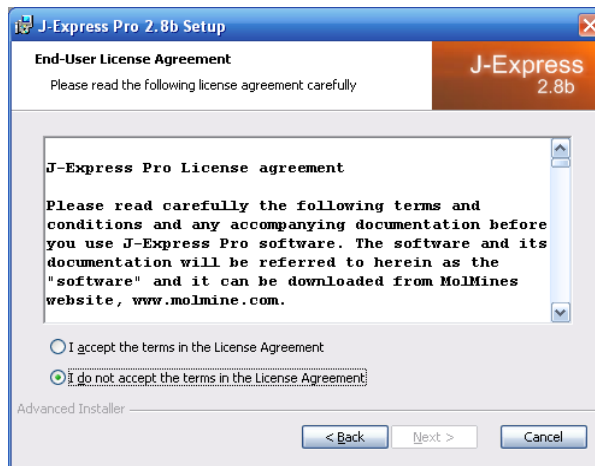
Accepting the J-Express license

After reading and accepting the licenses, enter the path of the directory you wish to install J-Express Pro to, or click "Browse" to locate it. If you enter a path to a directory that does not exist, the installer will, if possible, create it for you. Click **Next>** to continue the installation process. The required files for J-Express Pro will now be copied to your hard drive.
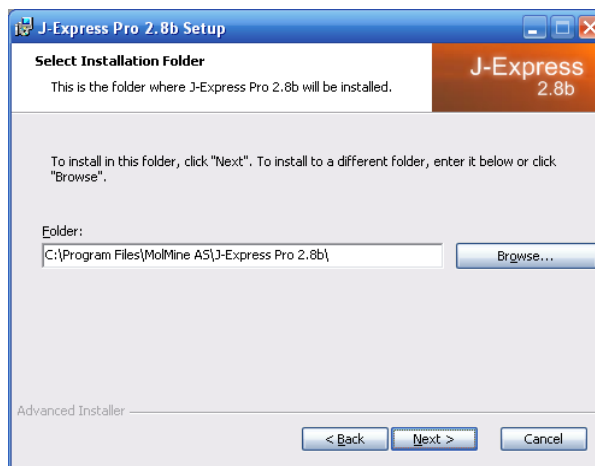
Installation Path screen

The installer displays the chosen installation path in the next window that appears, to allow you to verify it. Click **Next>** to keep these settings and start copying files to the installation directory. After the file copying is done, click the **Finish** button in the window that appears to complete the installation process.

2.2 Introduction to J-Express Pro

The first time you start J-Express Pro a welcome message is displayed. Close this window to start using J-Express Pro.

The J-Express Pro Desktop
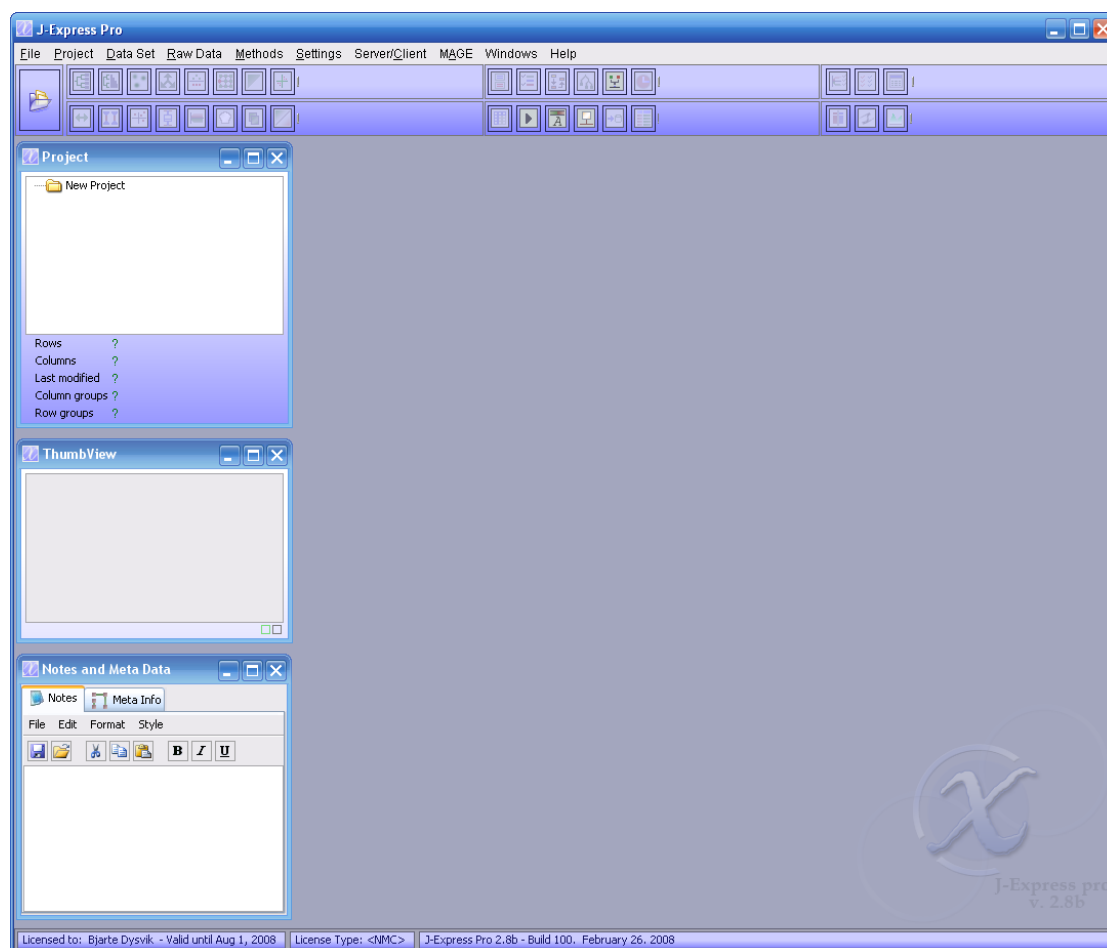
LICENSE KEY

If you have received a license key from MolMine AS, you must put that key in the J-Express folder (where you installed J-Express). If you start J-Express without a license key, the framework will start with a default dataset so that you can see and try the various methods. This preview mode does not allow you to load your own data. *If the license key is present in the J-Express folder, you can load your own data.*

The J-Express Pro interface consists of three parts. Along the top of the window there is a menu bar with pull-down menus. Just below is a toolbar giving quick and easy access to some of the advanced features of J-Express Pro. Along the left side of the window (from top to bottom) are the project management, thumbnail chart, and Info/Metadata windows.

The large blue area on the right side of the main window is used for displaying and managing data, analysis results, and dialog windows, through various sub-windows.

In this introduction you will be guided through the most commonly used features of J-Express Pro. For complete descriptions of the various features of J-Express Pro, please refer to Chapter 3 of this manual. Note: If you are using the demo version of J-Express Pro, all save/load/export functions on datasets will be disabled.

2.2.1 Loading Gene Expression Data

Data can be loaded into J-Express Pro from files formatted in many different ways. For this introduction we will be using an example dataset included with J-Express Pro.

If you have already saved the data as a .pro file, you may drag this file onto the project tree to load it.

1. Click the icon on the tool bar, or click **File** on the menu bar. Select **Load Tabular Data** from the menu that appears.

2. Click the **Manual** tab in the data loader window that appears to give you direct control of how data is imported to J-Express Pro. Click the icon. To bring up a file selection dialog where you can choose the file you want to import the data from. Locate and select the file `TutorialData.txt,` and click **OK**.

3. J-Express Pro allows data to be imported from files where the data columns are delimited either by tabulator marks or by simple spaces. Our file is a tabulator-delimited data file so select the radio button marked **TAB** in the dialog that appears, and click **OK**.
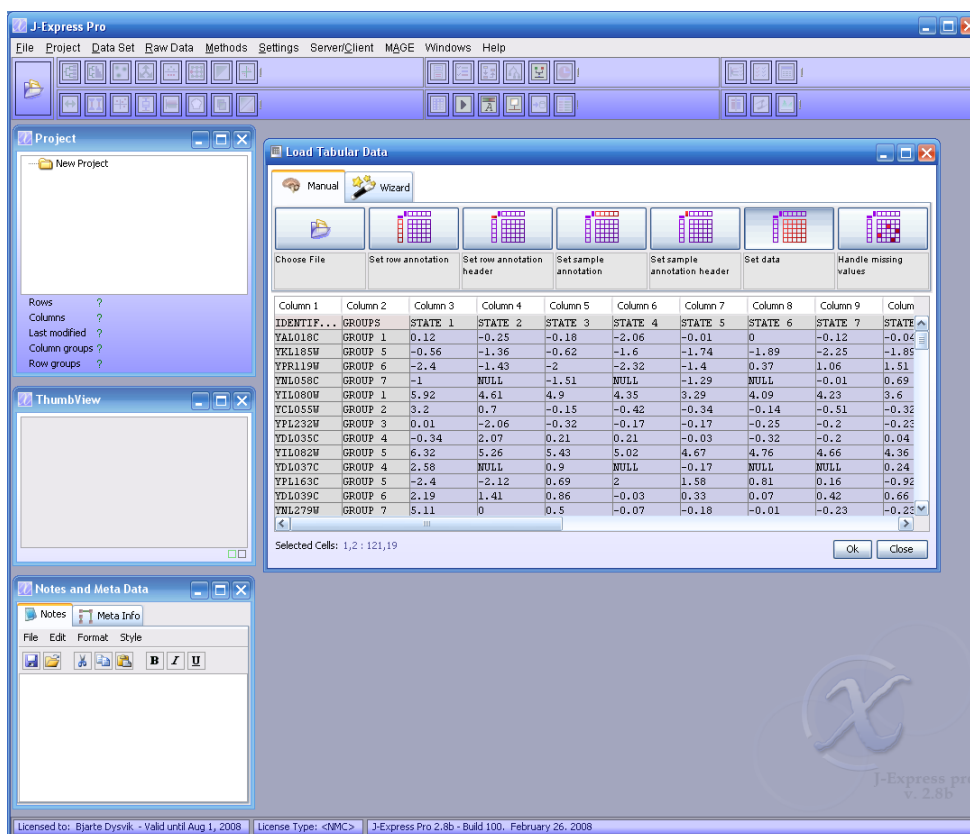
The data loader window allows for flexible data importation.

4. The data will appear in the data loader window. To set external information on the rows (e.g. functional groups), click the **Info** button, and select the appropriate columns. In our example, columns A and B contain the external information. J-Express Pro supports multiple columns of external information, if needed. The column(s) containing the external information turn gray when selected. To set external information columns is optional.

5. Click the leftmost **Info Headers** button from the left to select the cell(s) containing the headers for the information columns you selected in the previous step. Drag to select multiple cells.

6. Click the **ID Rows** to select the row containing the column identifiers. In our example this is the uppermost row, so click on any cell in this row to select it. The row then turns gray to indicate that it has been selected.

7. The tutorial dataset does not contain any row headers. To select the cell containing the row headers in a dataset containing such cells, click the rightmost **Info Headers** button, and then click the cell containing the row headers.

8. Click the **Data** button to set the cells containing the actual data. Click the upper leftmost cell containing a numeric entry (with value 0.12). Then scroll to the lower right cell using the scrollbars. Hold down the Shift key on the keyboard and click the lower right cell (with value –0.15). All the cells between the upper left and lower right cells are now selected as cells containing data. This is indicated on the spreadsheet by a blue color.

9. If you examine the values in column D (state 2) you will notice that two of the values for this state are missing. J-Express Pro allows you to manually correct these missing values by double-clicking on the cell with an erroneous value and enter a new value. If there are a lot cells with missing cells the alternative is to use the missing values dialog. Click on the **Missing** button to bring up this dialog. Make sure the radio button "Row average" is checked, and then click "**OK**". This will replace the missing values with the average of the values lying on either side of the missing value in the row.
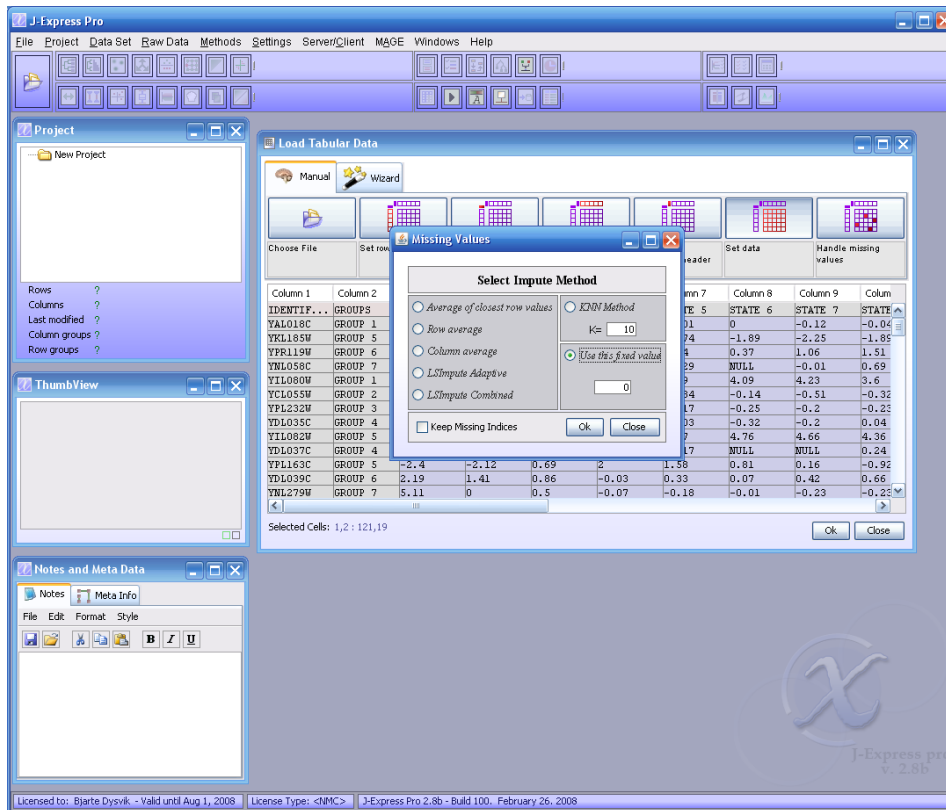
The missing values dialog helps replace missing data.

J-Express Pro is now ready to import the external data. Press the "**OK**" button to import the data and close the Data Loader Window.

Refer to Section 3.1.4 for information on how to load image analysis output files, and prepare these data for analysis using J-Express Pro.

2.2.2 The Project Tree
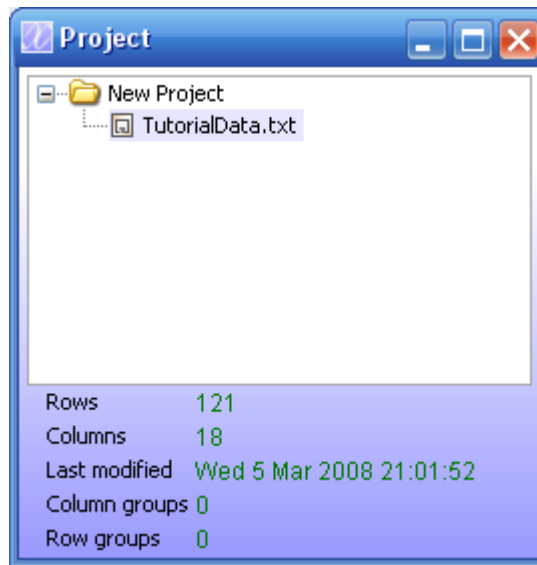
The project tree helps organize your data.

The Project Tree window is located in the upper left corner of the screen. The project tree allows you to easily keep track of the data files and data sets in a project, and to select and export a subset of the data.

1. Double click the folder named "New Project" in the project tree. A node named "TutorialData.txt" is shown below the "New Project" folder. All node names can be changed into whatever you like by double clicking on the **text** to the right of the node you want to rename, and then entering the new name. Click the node named "TutorialData.txt". The thumbnail window below the project tree window will be updated, and shows all the data in the dataset. By clicking the small squares below the chart, you can either show all profiles in the dataset or just the selected ones. Additional information about the dataset is available in the window below the thumbnail chart. On the **User Info** tab you can enter notes about the project, which will be saved with the project and reappear the next time you start working on it. Try entering some text into the Info text area. The meta info is automatically generated by J-Express Pro and will show all processes and methods applied to a dataset.

2.2.3 Hierarchical Clustering

1. Make sure the "TutorialData.txt" node in the project tree is selected (select it by clicking on it once). Click the ⊞ button on the toolbar. This starts the computations needed to produce a hierarchical clustering of your data set, and first opens a window that allows you to select the distance measure and linkage method to be used. Click **Ok** to use the default options. When the computations are completed, a Hierarchical Clustering window opens. Try pointing the mouse on a branching point in the tree. The sub tree defined by this branch will be highlighted red. The missing (null) values that were interpolated as described above will appear as blue rectangles. Positive values appear as red rectangles; negative values appear as green rectangles. Dark colors indicate relatively low values, and bright colors indicate high values. These colors can be changed to suit your needs.

2. Use the controls at the bottom of the window to change the dendrogram appearance. If the size is big enough, you will also see the annotation to the left in the window.
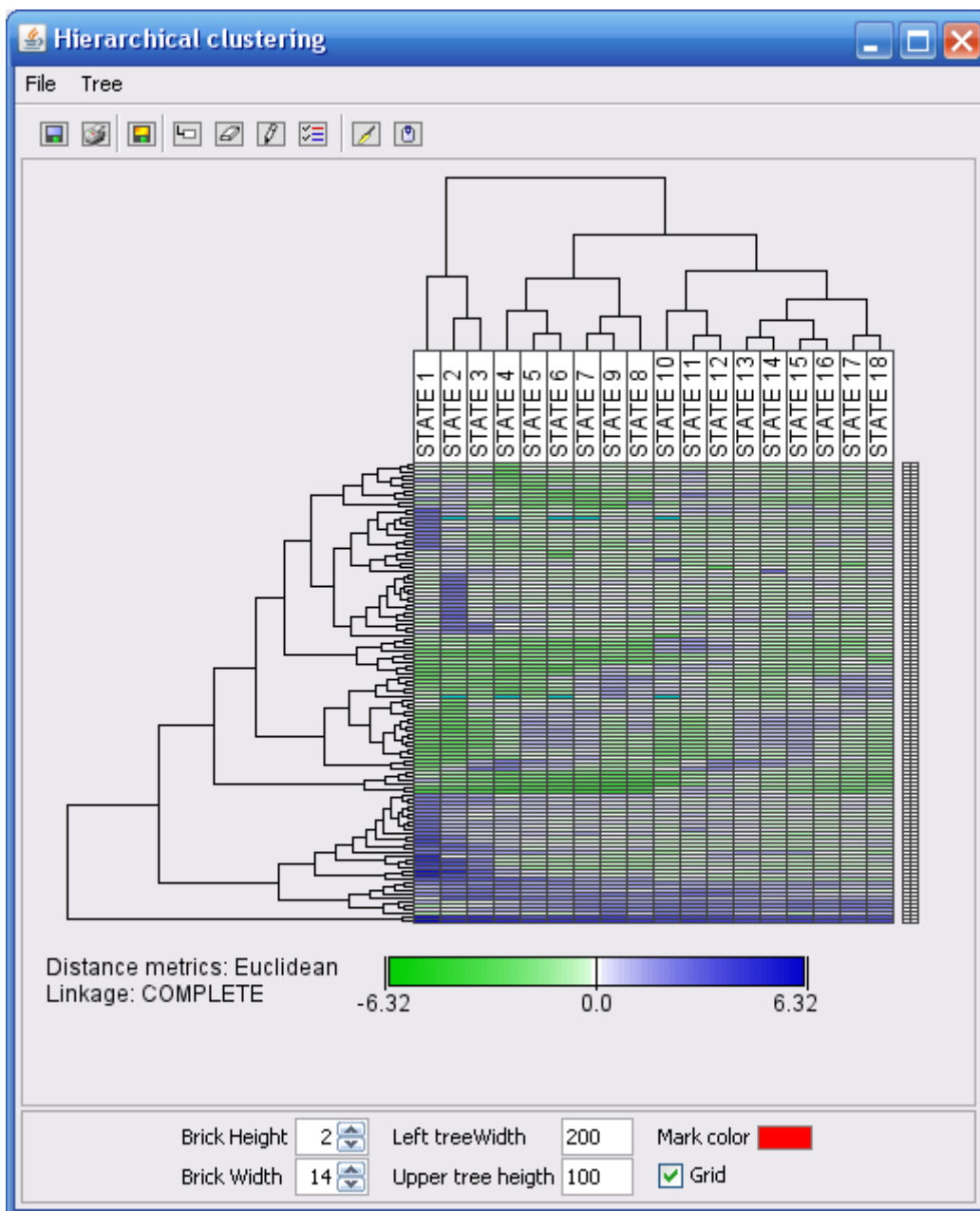
A dendrogram of a (zoomed) subtree.

3. Point on a connection (branch) in the left tree and watch the selection be updated to the right. To create a new node in the project tree that contains only the data (genes) selected, make sure the ⬚ button on the toolbar of the Hierarchical Clustering window has a green frame and click the left mouse button. The new node will be labeled "Branched", but you can change this label by double-clicking it and entering the new label. The new node is a subset of the parent node, and contains the same data as the dendrogram you branched.

2.2.4 K-Means Clustering
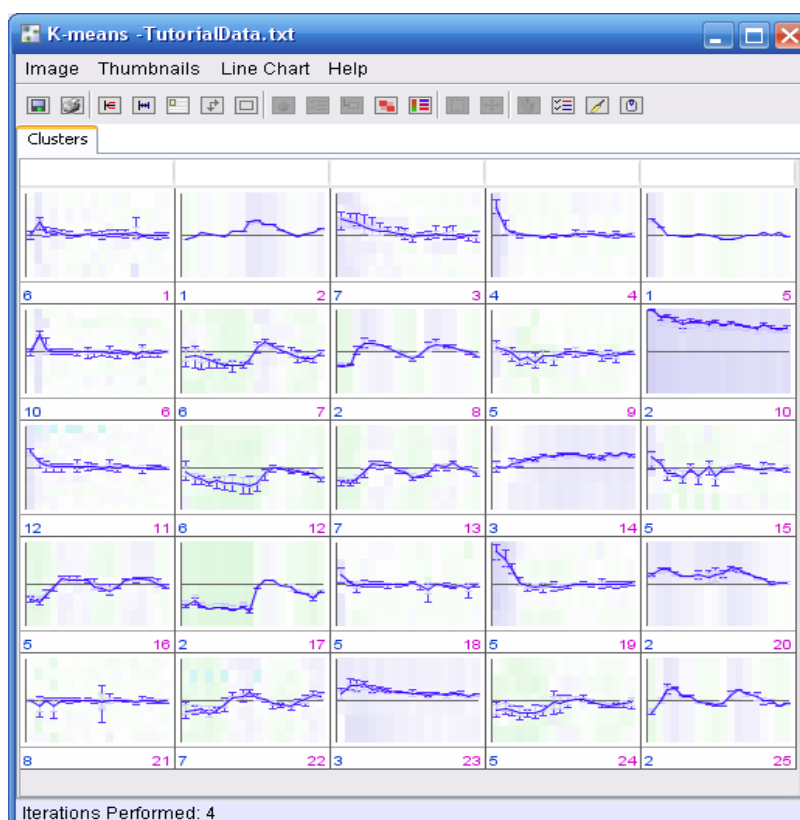
K-Means analysis with 16 clusters.

1. Make sure "TutorialData.txt" node in the project tree is selected. Click the ⬚ button on the toolbar, and then click **OK** in the dialog box that appears (to use default parameters). A K-means window appears showing thumbnails of the means of the clusters. To display all the profiles in a cluster click the ⬚ button on the toolbar of the K-means window. If you want to go back to displaying the means click the ⬚ (**Show mean profile**) button. To create a smoother chart click the ⬚ button to anti-alias the charts (gives higher graphical quality).

2. Click the ⬚ button to automatically generate a HTML file (web page) version of the K-Means analysis. An image folder containing the thumbnail images will be saved together with a HTML file with the name you input in the dialog that appears. Make sure you give the file the suffix ".html" (e.g. `myKMeans.html`), so that your web-browser will be able to recognize the file.

3. Click on one of the thumbnails in the K-Means window. A new tab labeled with the ID of the cluster you selected is added next to the "Clusters" tab. Click on this tab to display the selected cluster in a line chart window. For an introduction to the features of the line-chart window (gene graph viewer), see Section 3.2 of this manual.

2.2.5 Principal component analysis (PCA)

1. Make sure the "TutorialData.txt" node is selected in the project tree. Click the ⬚ button on the J-Express Pro Toolbar to open the PCA window.
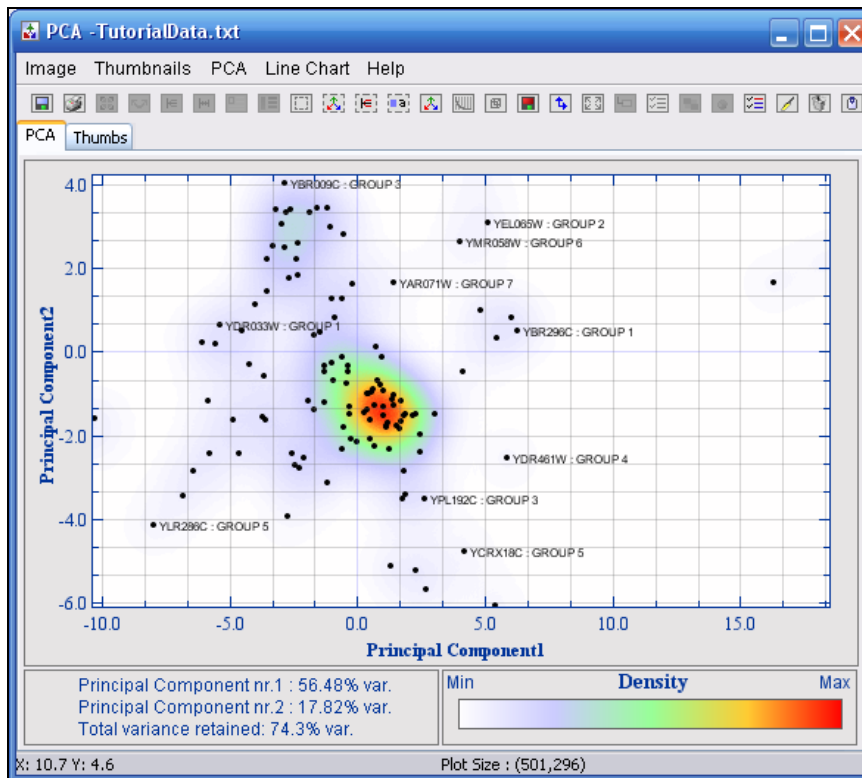
PCA Analysis window with density background.

2. To focus on a group of points you can use the selection tool. Press the button on the PCA window tool bar to enter frame contents to chart mode (⬛) , then click the ⬛ button and select the square selection method. Drag out a selection rectangle, and make sure you include some points within the rectangle. If you do not include any points in the rectangle, nothing will happen. After selecting an area from the PCA diagram a line chart thumbnail is displayed containing the profiles represented by the points you selected. Click the PCA tab again and select some more points. A new thumbnail will be generated. You can add as many selections as you like. The buttons that are active on the toolbar while on the "Graphs" tab have the same functionality as those described in Section 2.2.4.

3. Click one of the thumbnail charts. A new tab will be created named "Zoomed 1" containing a full line chart version of the selection the thumbnail represents. Click this tab. For an introduction to the features of the line-chart window (gene graph viewer), see Section 3.2 of this manual.

4. Select the PCA tab again, and then click the ⬛ button on the PCA window toolbar. A gene graph window is displayed, with profiles that may appear like random profiles, at first. This is actually the components upon which the data has been projected. To make sense of this chart, click the ⬛ (Shadow Unselected) button on the toolbar. Now select for instance components 1-3 by clicking on component 1, holding down the shift key on the keyboard, and clicking on component 3. The three selected principal components are displayed clearly on the chart, while all the others are painted a shade of grey. Similarly, you can select non adjacent components from the list by holding down the control key on the keyboard while selecting components.

5. Close the Gene Graph window. Select the PCA tab, and click the ▣ button. This creates a new tab labeled 3D. Click this tab.
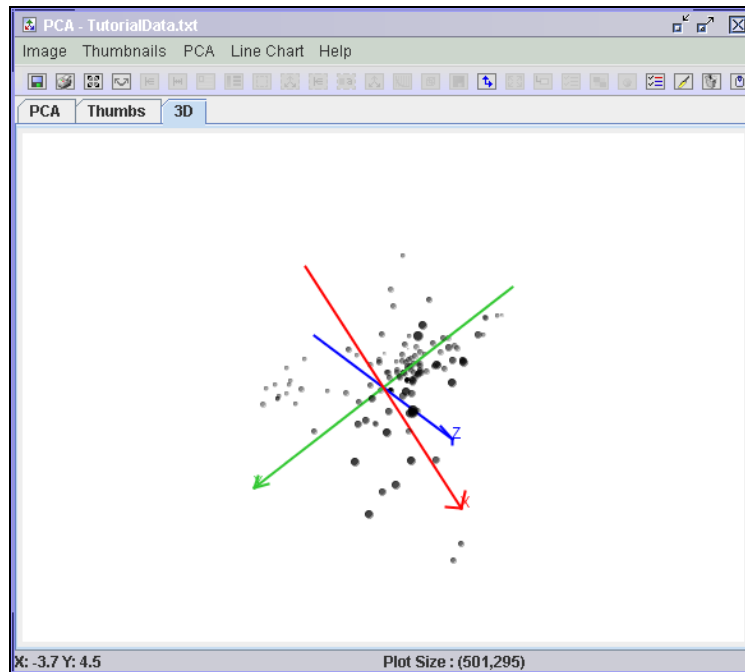
A three-dimensional view of PCA

The window shows a representation of the distribution of the PCA points in 3-dimensional space. To rotate the viewpoint simply click and drag with the mouse in the window.

6. Go back to the PCA tab again. Right-click in the PCA window to access the PCA properties window. Select Density Map from the Fill menu in the dialog that appears. Enter a value of about 50 for in the Paint Threshold box in the Density Map options area that appears, and click **OK**. Notice that dots in the densest areas of the PCA diagram have disappeared. When datasets are large, you can use this feature to prevent dense areas becoming black clouds of points, or to find points at the outer edges of the dense areas.

7. Bring up the PCA properties window again by right-clicking anywhere on the display area of the PCA window. Click on one of the colored squares to select a new color in the dialog that appears and then click **OK**. Click **OK** in the PCA properties window, and notice the changes in the density map of the diagram.
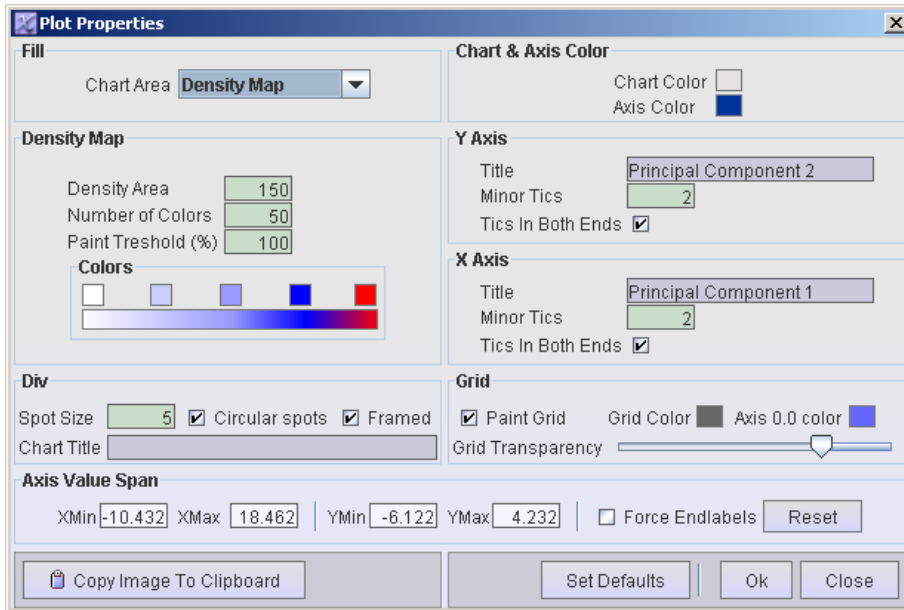
The PCA Properties window allows you to customize all aspects of the PCA diagram.

8. Go back to the PCA properties window again. Try out some of the other options available to you. Change the size of each spot by entering a larger or smaller number the Spot size text field. Click **OK**, and note the effects of your changes. You can also choose whether or not to display the various statistics and density scale by checking or unchecking the appropriate boxes.

9. Close or minimize all the open windows in J-Express Pro to prepare for the next part of this introduction.

2.2.6   Self Organizing Map (SOM)

1. Make sure that you have the "TutorialData.txt" node selected in the project tree, and click the ▦ button. Select 25 neurons and click OK
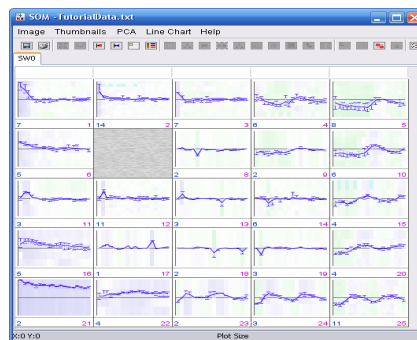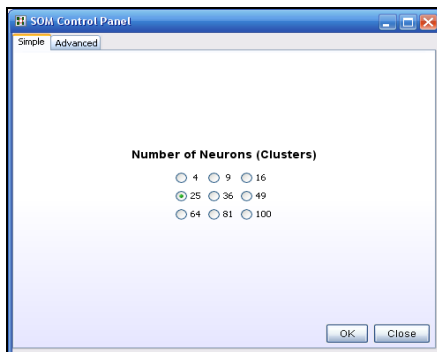
The SOM Properties window gives the user full control of all aspects of the generation process for self organizing maps.

2. Click one of the clusters to get details about the cluster members. When clicking a cluster, a new tab will appear next to the SW0 tab. Click the tab to see the details.

2.2.7 Gene Graph viewer

1. Make sure the "TutorialData.txt" node is selected in the project tree. Click the button on the toolbar to bring up the Gene Graph viewer, showing all the profiles in the TutorialData.txt set in the same chart.

2. If your computer is connected to the Internet, click the button to bring up the external link list. This adds a new list on the left part of the Gene Graph with the same content as the profile list. Select a profile in the upper list. The same profile will be selected in the lower list. By double-clicking the profile in the lower list a web browser will be opened (if necessary) and do a search for the selected profile in a public database. To use a different database, or add a new database, see Section 2.2.10.

3. Click the button to hide the external links list. Click the button, (Shadow Unselected) to shadow all profiles but the selected one. Click the button to automatically generate a HTML file (web page) version of the Gene Graph. An image folder containing the thumbnail images will be saved together with a HTML file with the name you input in the dialog that appears. Make sure you give the file the suffix ".html" (e.g. myGenegraph.html), so that your web-browser will be able to recognize the file.
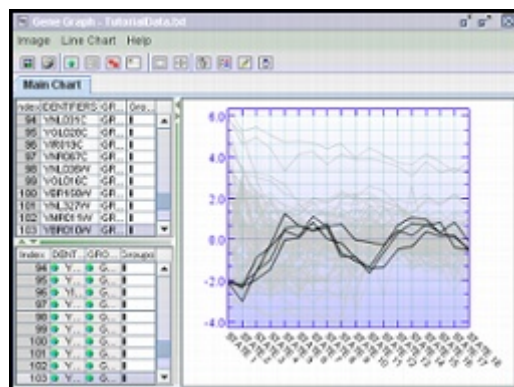
Gene Graph viewer with external link list.

2.2.8 Search and sort

Make sure you have the "TutorialData.txt" node selected in the project tree. Click the button (Search and sort) on the J-Express Pro toolbar. In this window you can search for values, annotations and sort annotations or samples. Type a string into the search phrase field and press enter. For instance, type *YMR.** to search for all samples containing YMR in the annotation. In the result table, click a column header to search all rows in regards to the values or annotation in this column.
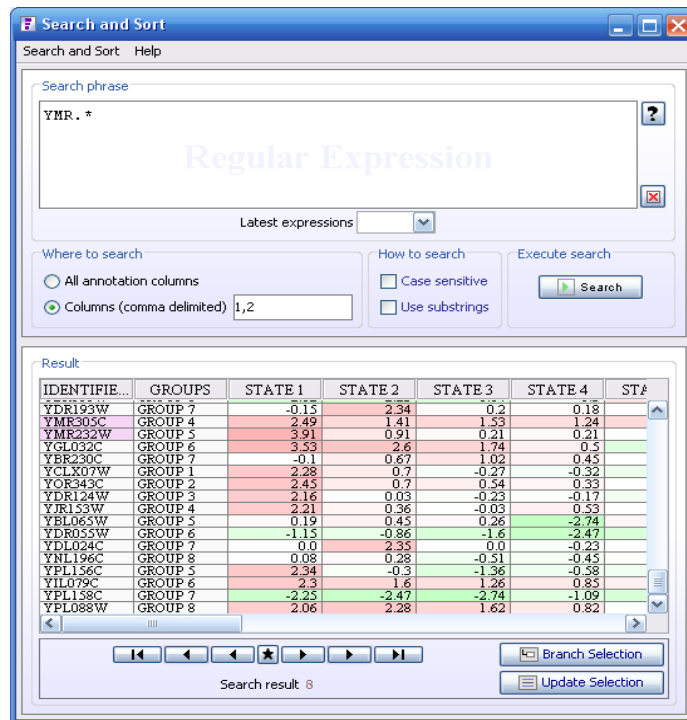
2.2.9 Finding Similar Profiles

1   Select the "TutorialData.txt" node in the project tree, and click the ⟷ button on
     the J-Express Pro toolbar. A list of all the profiles in the current node in the
     project tree is shown in the leftmost window. Select one at random. Drag the
     Tolerance (%) slider and select the 10 % closest neighbors. Check the **Update
     on Change** box in the upper right part of the window. Drag the **Tolerance** slider
     to 40 % and notice the way profiles are added to the display as you drag the
     slider.
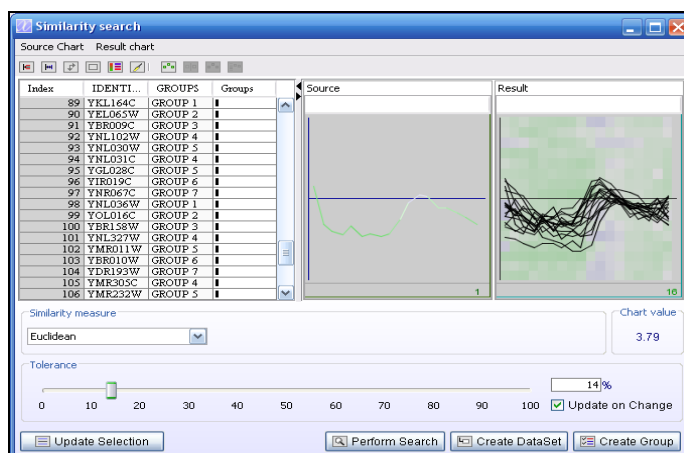
2   To create a new dataset based on the result of finding similar profiles, click the
     Create Dataset button. The new dataset becomes a sub-node of the
     "TutorialData.txt" node. Click the ⬥ icon next to the "TutorialData.txt" node in
     the Project Tree to display the newly created node.

2.2.10 Customizing the External database links

- Select the "TutorialData.txt" node in the project tree. Click the [image]button (**External Link List**) on the J-Express Pro toolbar or select **Methods |** [image] **External database links**. Click the URL List in the Web Resources window. This brings up a list of all the external databases that are currently accessible from J-Express Pro.

- To select a different database for profile lookups, click on another database in the list.
- To create a new link, click the Manage Links button in the Web Resources window. In the URL List window, click Add. The next part can be a bit tricky if you are not familiar with web scripts used with database searches. Since databases work in different ways, it is not straight forward to explain how to do this. Here is an of example:
  - ○ We can create a link to Yeast Genome Database. Open the page http:// www.yeastgenome.org/ and search for "JEID" in the search field. This opens a page displaying the search result. Copy the url to an empty row in the Link URL column in the URL List window. Add ? query=<JEID> to the end of the url. The address bar of your browser should now read something like this: http://db.yeastgenome.org/cgi-bin/SGD/search/quickSearch?query=<JEID>

- In the empty **Link Name** cell, type the name you want for the new search, for instance "The new search", and press enter. Click Save and close the window.
- Some databases can search for several genes at one time. Each gene is then separated by a Selection divider. The selection divider is often & (and) or | (or). This should also be specified on the database help pages. Type the selection divider to use in the Selection divider column.
- Test the new external link by selecting it from the URL List menu and clicking a profile in the Web Resources window. A page with the search results should open in your web-browser.
- The process for other online databases is similar. Use the database help pages to find out how link up to that particular database.
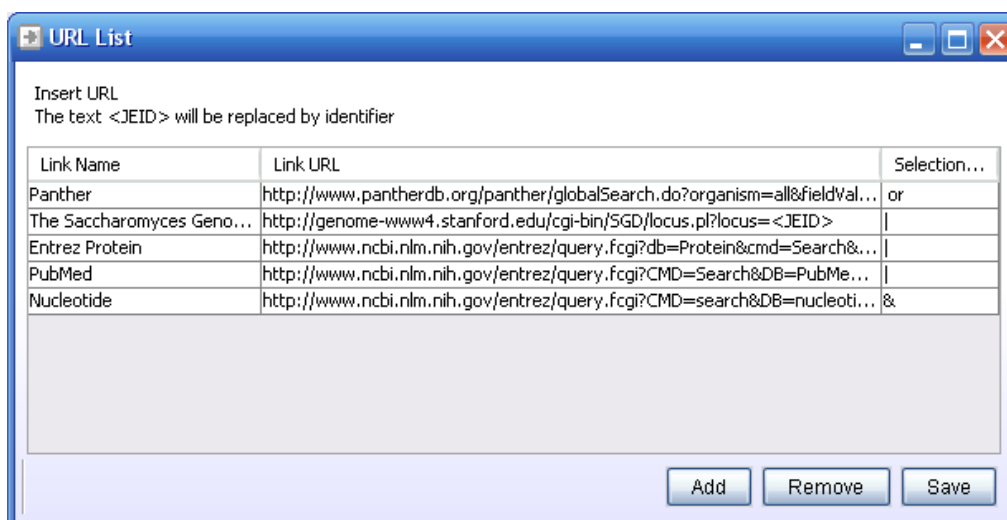
2.2.11 Creating and managing groups

1. To create a group, select the "TutorialData.txt" node in the project tree and click the ⊞ button (**Create Groups**) on the J-Express Pro tool bar. Type GROUP 2 into the **Selection String** text field and press enter. Use the scrollbar on the list in the middle of the window to verify that only profiles from group 2 are selected.
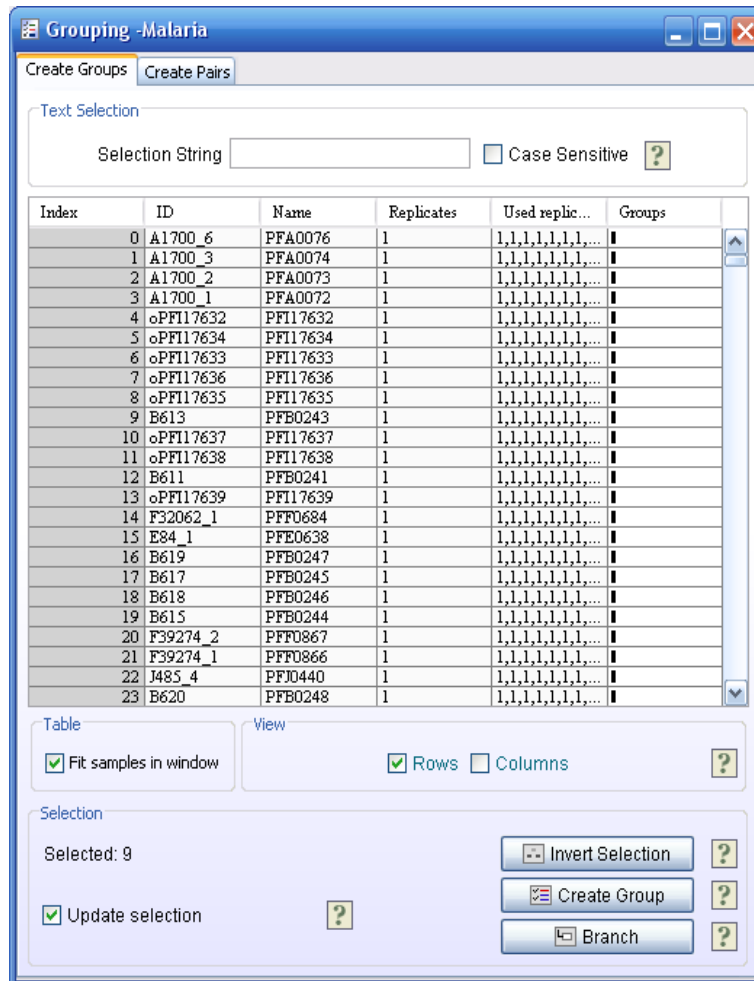
The Create Groups window.

2. Click the Create Group Button, choose a red color, call the group "Group 2", and click **OK**.

3. Delete the text in the **Selection String** text field. Scroll to the top of the list and drag the mouse over the upper six rows to select them. Click the Create Group Button, choose a blue color and click **OK**.

4. Close or minimize the window.

5. Click the ⬚ button (Open Group Controller) on the J-Express Pro toolbar. This brings up the Groups window, with 3 groups already defined. The upper one has no name. Double click the **Group Name** cell of this group and type

   `Upper  6`. Leave this window open, and click the ⬚ button (Hierarchical Clustering). Notice that members of the two groups are marked with red and blue boxes to the right of the value rectangles.
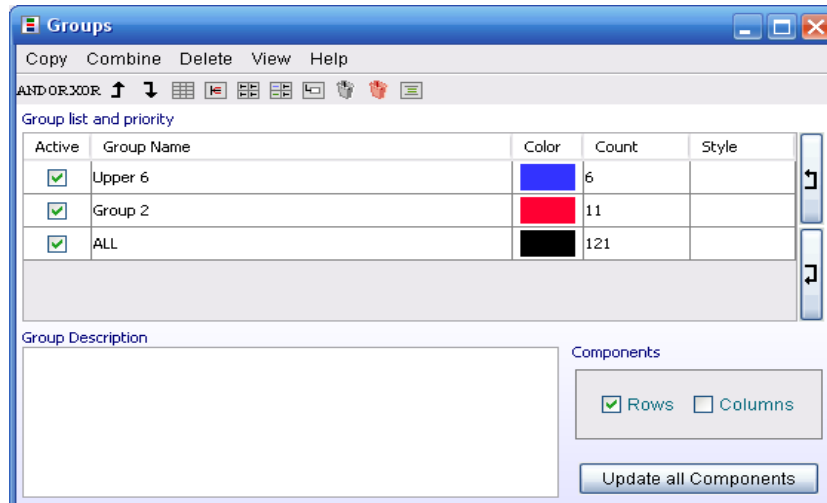
The Groups window provides an easy way to create and manage groups of data.

6. Leave the dendrogram open and click the ⬚ button (Principal Component Analysis). On the PCA diagram the points belonging to a group is marked with the respective colors of the group. Click the ⬚ button (**Frame to** Chart), and select an area by dragging the mouse over some of the dots belonging to a group. Do this again to create another thumbnail chart. Click the ⬚ button (**Create Group(s)**) from the PCA tool bar. Two groups named Cluster 1 and Cluster 2 has now been added to the Groups window. You can edit the names of the groups by double clicking in the rows of the Group Name column. Click the color boxes for the new entries to assign a color of your choice to the new groups. Click the **Update all Components** button update all components with their new group colors. If you take a look at the open PCA windows and dendrograms you will see that they have been updated with the new groups automatically.

7. Uncheck the **Active** box for all groups except the two uppermost ones, and click **Update all Components**. If you bring back the PCA diagram window you will notice that only the points of the selected groups are displayed. Check the **Active** for all groups again, and click **Update all Components**.

2.2.12 Managing Projects

1. Close all open windows. Open the K-means clustering dialog by clicking on the

   ⬚ button on the J-Express Pro toolbar. Keep the default settings, and click **OK**. Click a few of the thumbnails to bring up some larger charts (select the new tabs). Select one of the larger charts, and click the ⬚ button (Branch Data), and notice

how the branched data is inserted into the Project Tree. This new node can then be analyzed further by using any of the functions of J-Express Pro just like a normal dataset. Double-click the label of the newly created node ("Branched") to give it a more appropriate label. Enter the new label, and press enter. Notice that the Icon for the new node matches the method the data was branched from. To remove a branched dataset from the Project Tree, right-click the dataset and select delete dataset.

2. To save the project click the button on the J-Express Pro tool bar to bring up the file menu. Select **Save Project As** and enter the filename `tutorial.pro`. Saving the project saves the entire Project Tree. Choosing **Save Tabular** saves all the data in a tab-delimited text file.

3   Reference - The Complete J-Express Pro Guide

3.1 Projects

All analysis in J-Express Pro is done within the context of a project. A project in J-Express Pro consists of a number of data files, notes, and meta data. The data files can either be "raw data" (output from image analysis programs) or "gene expression data", and the files can have many different formats. Notes can be entered in J-Express Pro by the user, and are saved with the project. The generation and maintenance of meta data provides an auto-documenting feature for the user of J-Express Pro, and is saved with the project for all data sets stored.

If you have saved data as a .pro file, you may drag this file onto the project tree to load it.

3.1.1 The J-Express Pro tables

All tables is connected trough a data-listening / selection change event set of interfaces. This means that changes such as selection changes in one of the tables (for instance the hierarchical clustering table) will also be made in other open tables visualizing the same dataset. If a selection has been made, new windows will also be updated to have this selection. You should use this feature to visualize results in different components. For instance, having found a selection of interesting genes in the hierarchical clustering component, select all indices in the table and open a gene-graph viewer. Now click the "shadow unselected" button and the selection will appear also in this component.

New projects
1. Select the **Project | New Project** menu item from the J-Express Pro menu bar.

By default, J-Express Pro starts with a blank project.
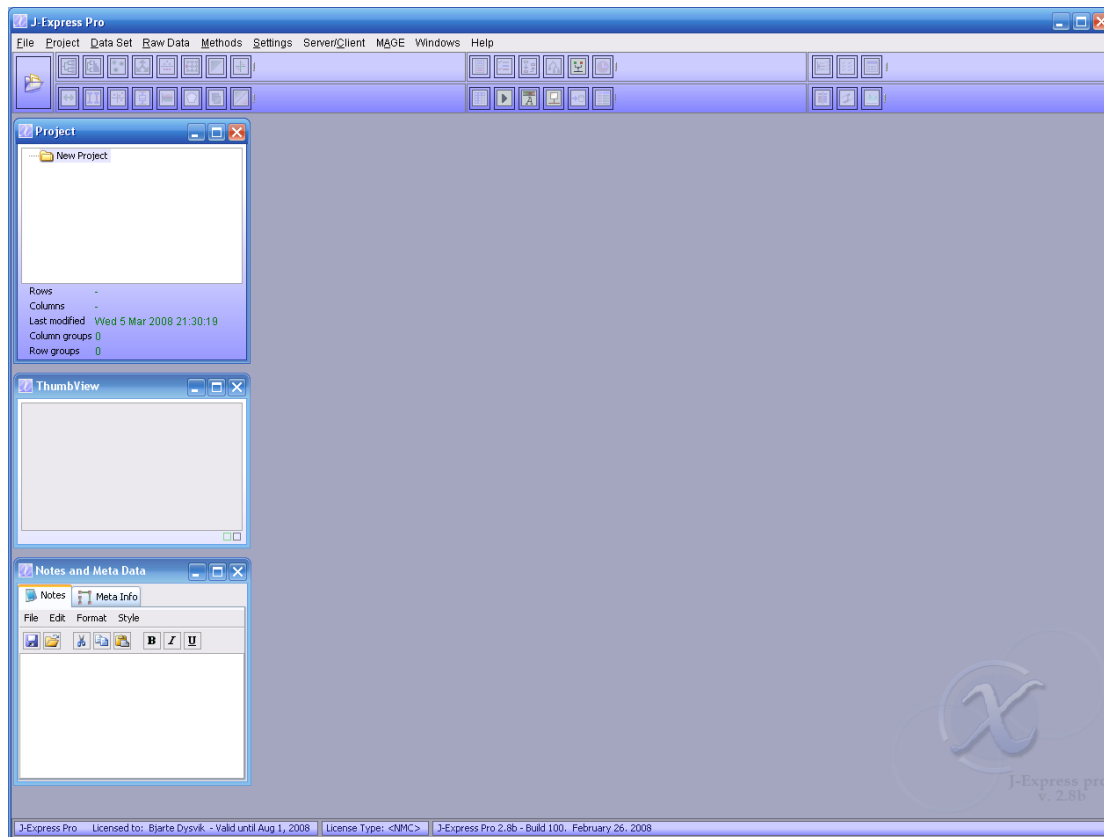
To change the name of the project from "New Project" to one of your choice double-click the label to the right of the ⬜ blue Project folder icon in the Project Tree window. Type the new project name and press enter.

J-Express Pro accepts data formatted in a variety of ways. The main requirement for data files is that it is contained in a text file (or in a set of text files), and that the data fields are delimited by either tabulator marks, or by simple spaces. J-Express Pro supports multiple columns of external (non data) information, and one cell of column identifiers in addition to a number of formats generated by common image analysis programs.

3.1.2 Importing gene expression data manually into J-Express Pro

1. Click the ✉ icon on the toolbar, or click **File** on the menu bar. Select **Load Tabular data** from the menu that appears.

2. Click the **Manual** tab in the data loader window that appears to give you direct

   control of how data is imported to J-Express Pro. Click the open button ✉. This brings up a file selection dialog where you can choose the file you want to import the data from. Locate the file containing your data, and click **OK**. An alternate way of loading data into the spreadsheet is to copy data from Microsoft Excel and paste it directly into the spreadsheet. In that case the next step is unnecessary.

3. J-Express Pro allows data to be imported from files where the data fields are delimited either by tabulator marks or by simple spaces. Select the appropriate choice for your data file and click **OK.**
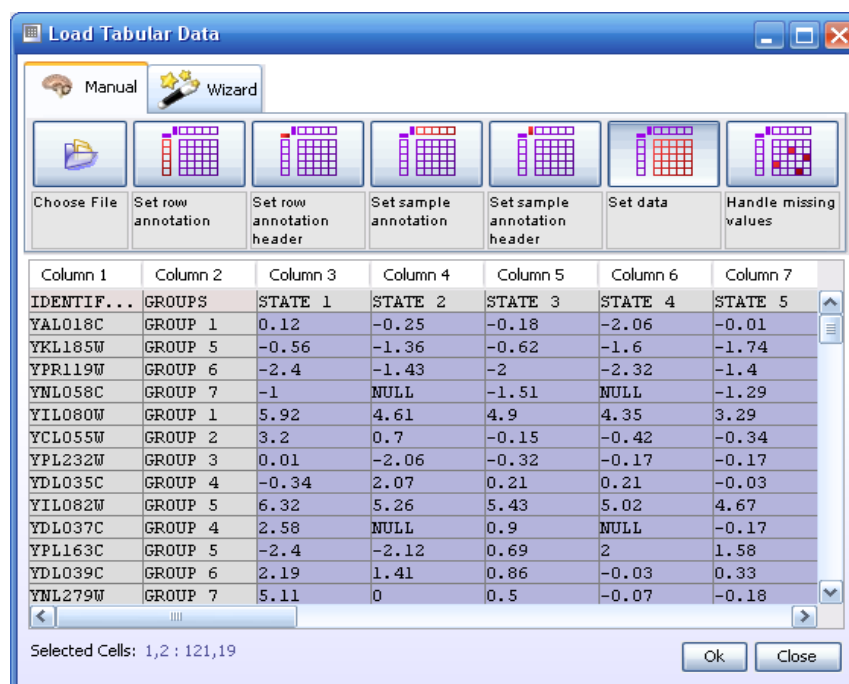
The data loader window after setting the identifier, information and data areas.

4. The contents of the data file will now appear in the data loader window. To set external information on the rows (e.g. functional groups), click the **Row Info** button, and select the appropriate column(s). J-Express Pro supports multiple columns of external information, if needed. The column(s) containing the external information are colored a shade of grey when selected.

5. Click the leftmost **Info Headers** button to select the cell(s) containing header information for the Info columns, and then click on the relevant cells.

6. Click the **Column Info** button to select the row containing the column identifiers. Click on any cell in the row containing column identifiers to select it. The row will be highlighted grey.

7. Click the rightmost **Info Headers** button to select the cell containing header information for the column identifiers of the previous step, if the dataset contains such information.

8. Click the **Data** button to set the cells containing the actual data. Click the upper leftmost cell containing a data entry, and then scroll to the lower right cell containing data using the scrollbars. Hold down the Shift key on the keyboard and click the last data cell. All the cells between the upper left and lower right cells will now be selected as cells containing data. This is indicated on the spreadsheet by a blue color.

9. Microarray scanning and quantization sometimes result in missing values in the dataset. J-Express Pro allows you to manually correct the missing values by double-clicking on the cell with an erroneous value and then enter a new value. This method usually becomes unwieldy in a large dataset. If there are a lot of cells with missing values, the alternative is to use the missing values dialog. Click on the **Nulls** button to bring up this dialog.
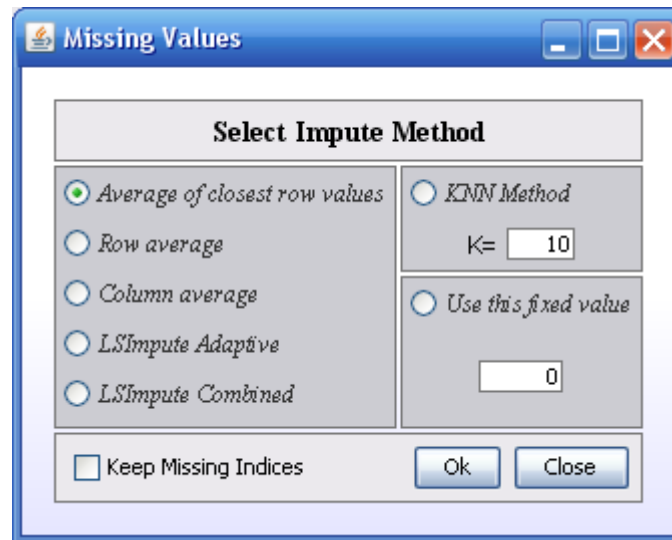
Select the appropriate method for replacing the missing values from your dataset.

- Average of closest values. Calculates the average value of the data entries to either side (if available) of the missing value, and then uses this average in place of the missing value.

- Row average. Calculates the average of all the data values of the row the missing values is a member of, and then uses this average in place of the missing value.

- Column average. Calculates the average of all the data values of the column the missing value is a member of, and then uses this average in place of the missing value.

- LSimpute Adaptive and LSimpute Combined - The Least Square impute methods exploit correlated genes to draw a best fit straight line y=ax+b through points representing the expression level of each sample. The idea is then that if the expression of gene x is known, the regression model can be used to estimate the expression level of gene y. Please refer to the following paper for method description:

  LSimpute: accurate estimation of missing values in microarray data with least squares methods
  Trond Hellem Bø, Bjarte Dysvik and Inge Jonassen,
  Department of Informatics and 2 Computational Biology Unit, BCCS, University of Bergen, HIB, N5020 Bergen, Norway.
  Nucleic Acids Research, 2004, Vol. 32, No. 3 e34

- KNN Method - It calculates the K most similar profiles based on Euclidian distance of the row containing the missing value, and then computes the missing value as the weighted average value of these profiles for the column containing the missing value. Please refer to the following paper for method description:

  Missing value estimation methods for DNA Microarrays.
  Olga Troyanskaya1, Michael Cantor1, Orly Alter2, Gavin Sherlock2, Pat Brown3,6, David Botstein2, Robert Tibshirani4, Trevor Hastie5, Russ Altman1
  1Stanford Medical Informatics, Stanford University School of Medicine Departments of 2Genetics, 3Biochemistry, 4Health Research & Policy and and Statistics, 5Statistics and Health Research & Policy, and 6Howard Hughes Medical Institute, Stanford University
  Bioinformatics. 2001 17:520-525.

- Fixed Value - sets all missing values to the value specified here.

J-Express Pro is now ready to import the external data. Press the "**OK**" button to import the data and close the Data Loader Window.

3.1.3 Importing Illumina data

The Illumina data import tool can be found under *file -> load illumina data.* With this tool you can load illumina data exported fom the illumina system as tabular format.
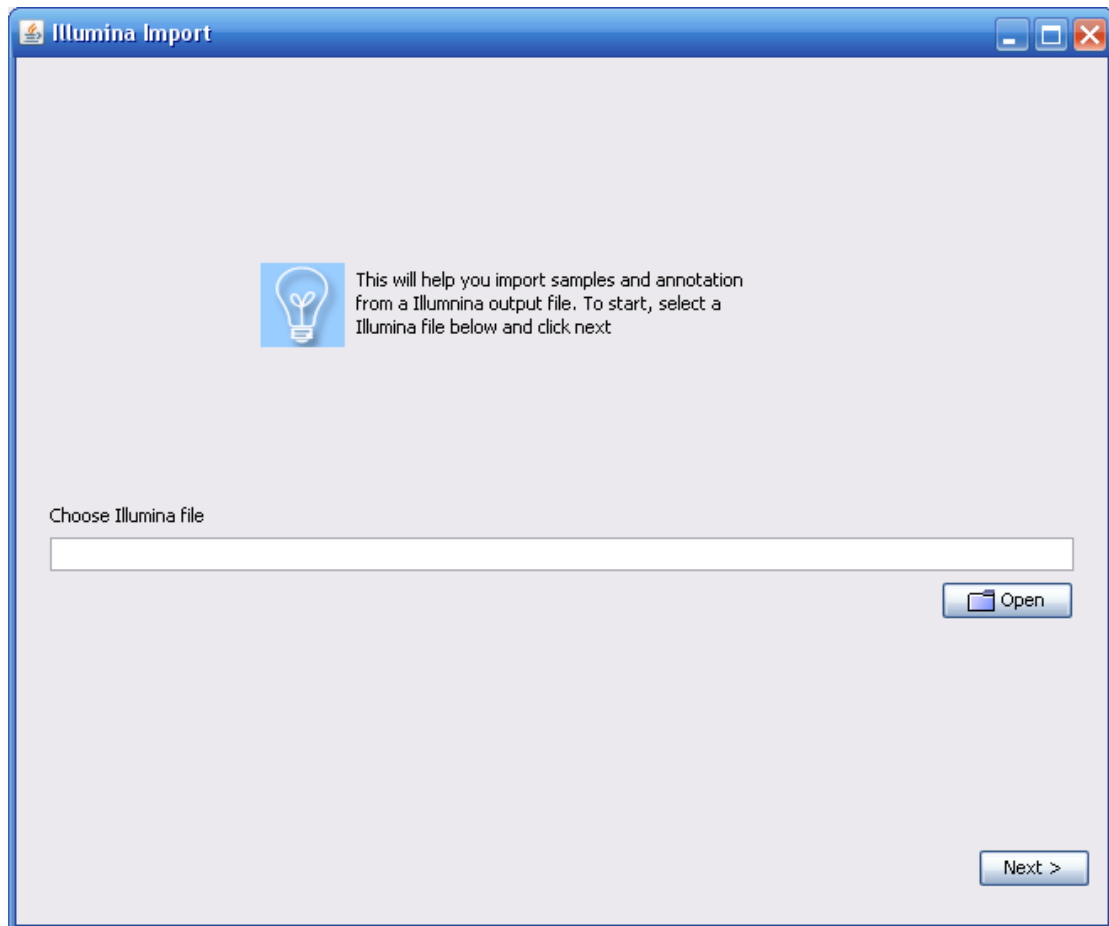
First, select the file you want to import. The Illumina data import tool requires all data for the experiment to be in the same file.
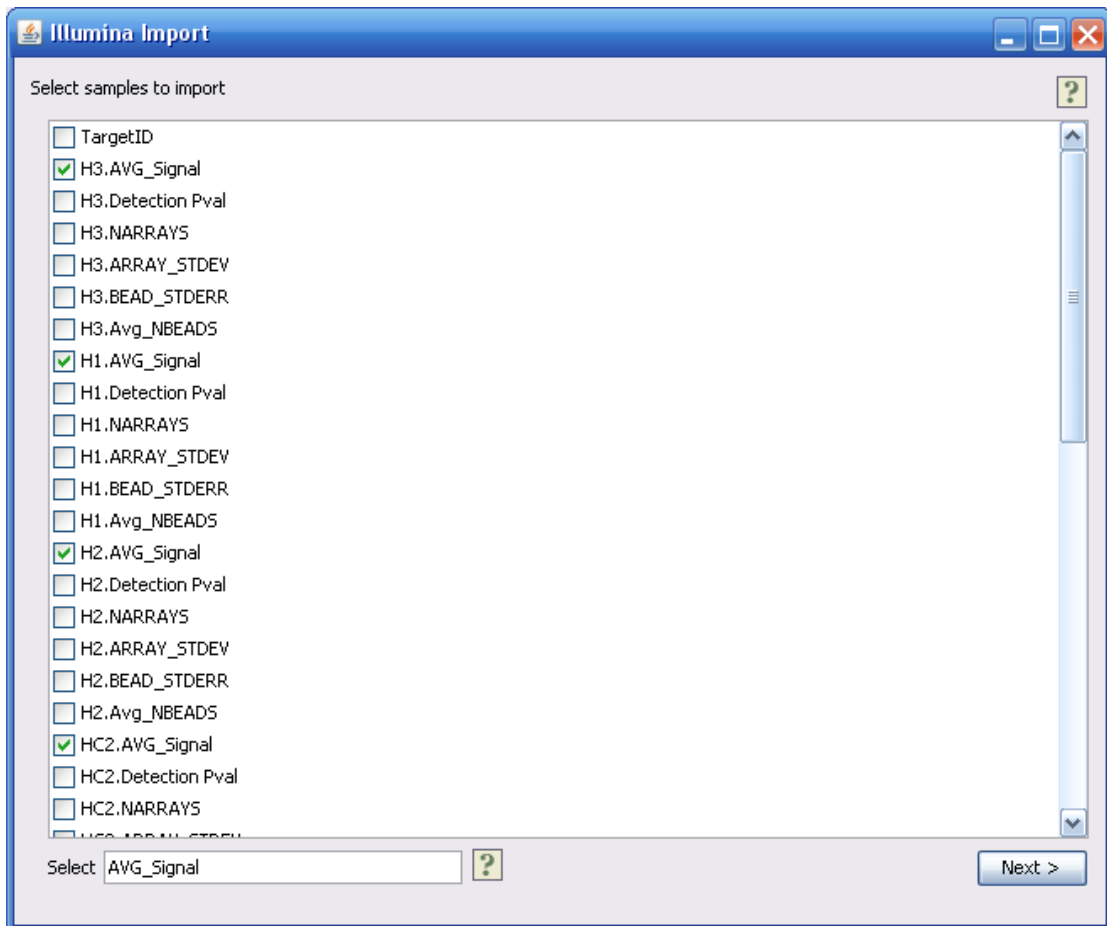
Next, select the columns to import to your dataset. Choose only the data columns to import, the annotation comes in the next step. If you type a string pattern in the *Select* field, you can choose many columns at the same time.
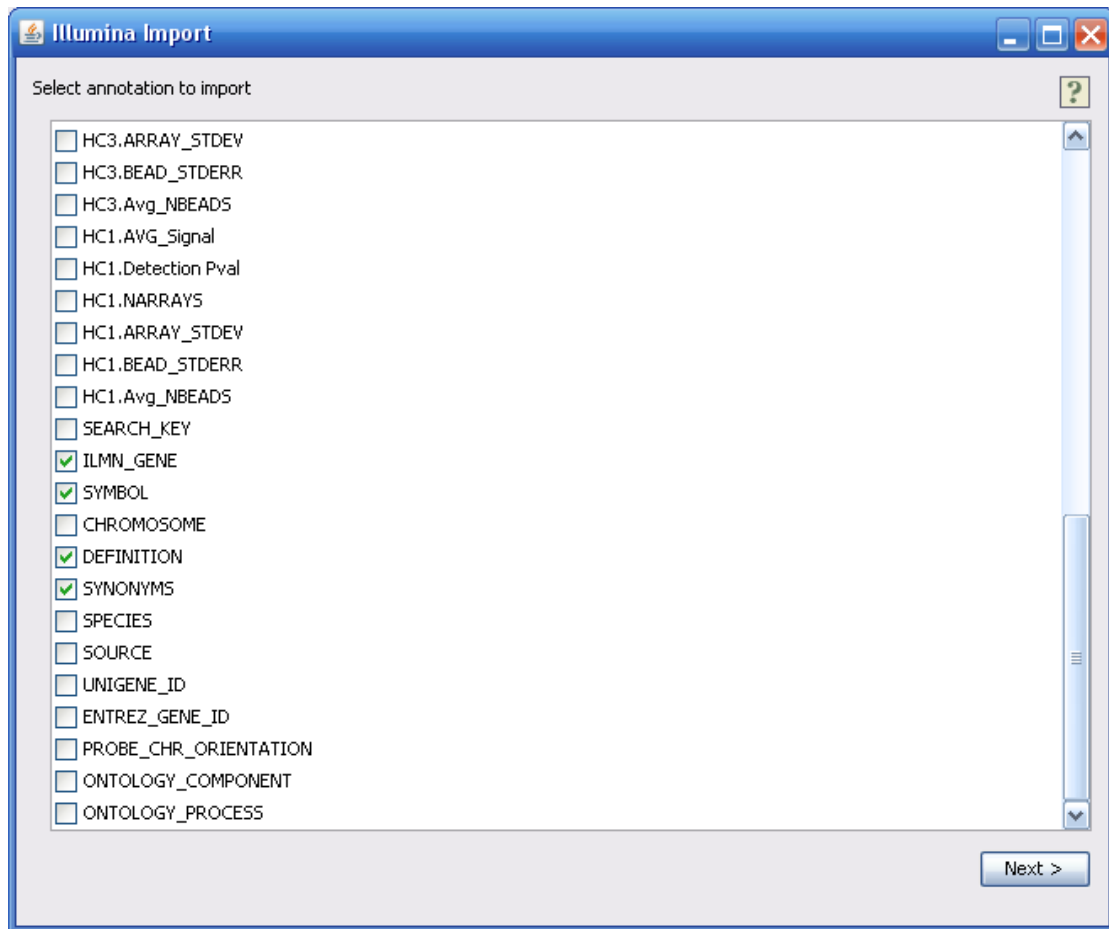
Next, select the annotation to import. Clicking next from this window will start loading the data, which may take a while.

The next windows shows the loaded data in various forms. The *data* tab shows the data in a table. The *plot* tab lets you choose two data columns to plot against each other. From the *processing* tab, it is possible to normalize the data with a quantile normalization method. The order of the samples can be changed in the *sample order* tab. It some of the samples share analysis groups, such as disease and normal, it is a good idea to put the samples from the same groups next to each other.

Clicking next from this window brings up the final window.
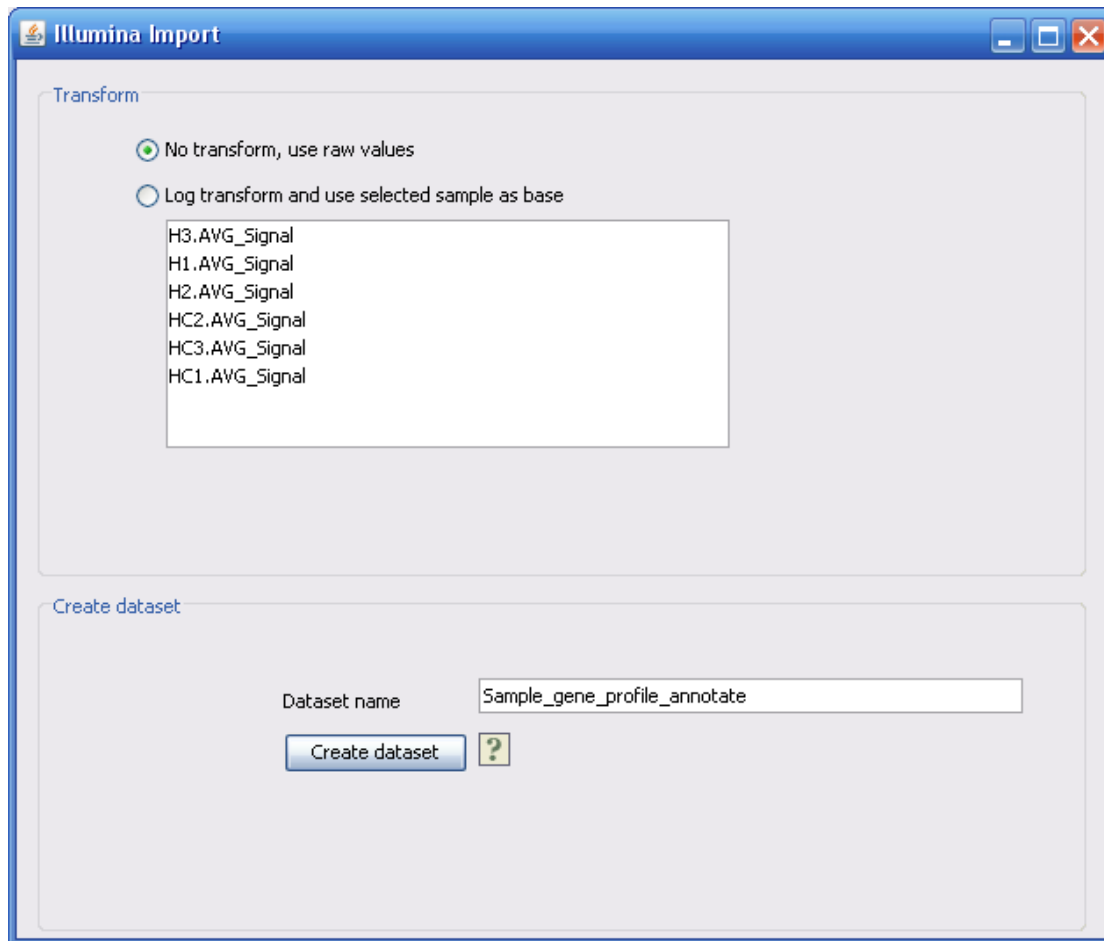
In the transform dialog, you can log transform the data with a base (if the data contains a base). Finally, specify a name for the new data set and click *create dataset.* The new data will appear in the project window.

3.1.4 Importing Spot Intensity (Raw) Data

J-Express Pro allows raw data from microarray analysis to be imported directly. If you have a format that is not currently supported here, you can specify your own formats in J-Express Pro.

To begin importing raw data into J-Express Pro select **File | Load Raw Data** from the J-Express Pro menu bar, or click the button on the J-Express Pro tool bar and select **Load Raw Data**. Alternatively you can click the **Open SpotPix Suite ( )** button from the J-Express toolbar or select **Raw data | Open SpotPix Suite**

This component is a framework for loading various forms of raw data. This data is normally filtered and normalized before an expression matrix is generated. If the data you want to load is already processed, you can use the load tabular data in the file menu instead.

Quick Start

If the data files you have is recognized by J-Express, you should be able to drag and drop the files from your file system onto the experiment table.
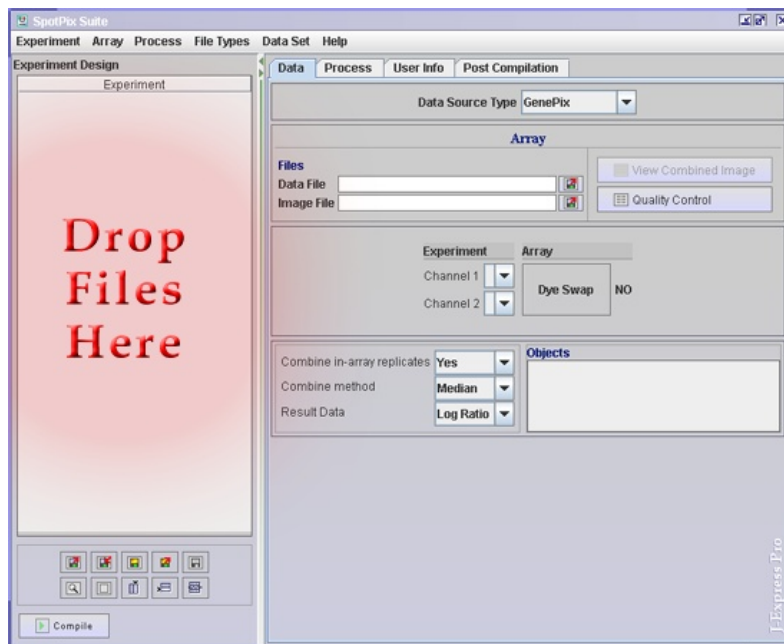
If the files are recognized, a set of default values will be selected. If the Data Source Type box does not change and still have GenePix Selected while your files are not from Genepix, you will have to manually explain to J-Express where to find the data in your files. Please see help on the Tabular data to continue. If the file is recognized, you may continue with experimental design and pre-processing.

Experimental Design

Underneath the experimental design block are some buttons. Locate the Add Experiment (⊞) button, and click it the same number of times as the number of arrays you have, (not included replicate arrays). Click the Add Replicate Column (⊞) button. This will add an array column. Right-click all cells in the Array column and choose Add Array. Double click all cells in the Experiment column and type in the name or identifier of each experiment. The last column contains ↓↑ arrows. To rearrange the order of the experiments, click and drag the arrows up or down to their new location.

Save Experiment (▦):

An experiment can at any point be saved by clicking the Save Experiment (▦) button. The experiment will be saved as a J-Express Pro experiment with the suffix .jex

Load Experiment (![icon]):

To load an earlier saved J-Express Pro experiment, click the Load Experiment (![icon]) button. A J-Express Pro experiment has the suffix .jex

Remap files to different folder(![icon]):

Sometimes you need to send project files to other people, who already have the data files. Since data files often are quite large you can remap the project files to the new folder instead of sending the all of the source files as well.

Load experiment from file list(![icon]):

Experiments can be loaded directly from a file list. A new row containing the array name, array will then be added to the experiment, and the file location set. This is a quicker way of adding arrays to the experiment than what was described above.

Reset File Location in Selected Dataset (![icon]):

This button resets the file pointers in the selected dataset. When a dataset is compiled, pointers to the data files are stored in the dataset object so that image spots can be extracted after data processing. Because it is not possible to change settings in a genepix project belonging to a dataset, it is in theory not possible to remap the file pointers in the dataset to a different location. This is however what this button does. If your data files are located in a different folder than defined when the dataset was compile, use the remap files to different folders, and then click this button to correct the pointers in the dataset selected in the project tree.

Check that all arrays are from same batch (![icon]):

The ![icon] button tests whether all arrays belong to the same experiment. Arrays belonging to different experiments can cause problems for instance if Combine in-array replicates has been set to no.

New Experiment (![icon]):

Click New Experiment (![icon]) button to clear the current experiment.

Remove selected experiments (![icon]):

To remove arrays from the experimental design list, select the rows of the experiment(s) you want to remove and click Remove selected experiments (![icon]).

▶ *Compile:*

When all arrays and processes (see section 3.1.4) have been added, click the ▶ Compile button to start processing the data. The processed dataset will be added to the J-Express Project Tree.

Linking the Datafiles

The data files that contain your experimental data has to be linked to each array image(

) in the array column (and replicate columns). First select the Data Source Type from the pull-down menu at the top of the Data Tab. Next click on each array image and set the file locations in the Data tab.

There are four different Data Source Types that can be selected:

- GenePix

- Affymetrix

- Tabular

- Project Dataset

3.1.5 Refining / Processing Raw Data

Most microarray raw data need further processing before analysis can begin. The processing steps involve filtering and normalization of the data. Select the array you want to process by clicking in the Array cell. Click the Process Tab. The Process Batch area holds all the processes you want to carry out on an array. The processes have to be added one at the time. Click the Add Process button.
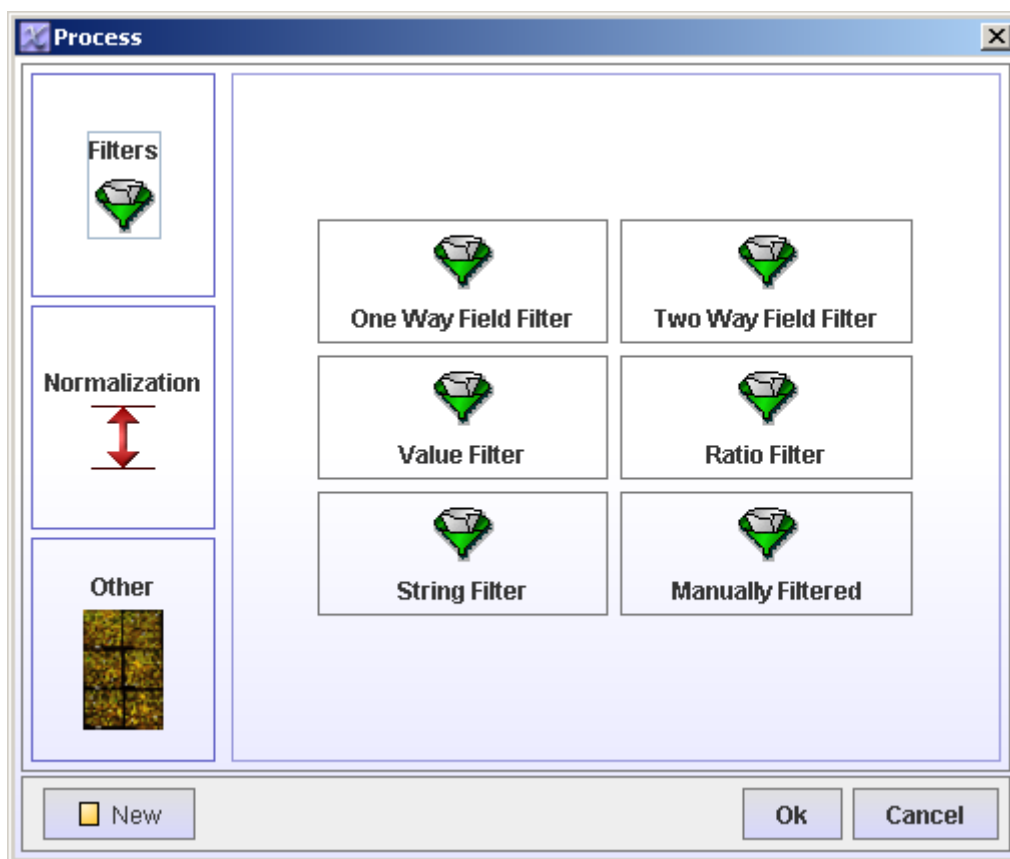
The Process window offers Filtering, Normalization and some other options.

Filtering

- One Way Field Filter - filters all spots with an attribute value above, equal or below a specified value.

- Two Way Field Filter - filters all spots with an attribute value above, equal or below 2 times the value of another attribute.

- Value Filter - filters all spots with a value above or below a specified value in at least one or all channels.

- Ratio Filter - filters all spots with a ratio above or below a specified value.

- String Filter - filters all spots with an attribute equal or not equal to a regular expression.

- Manually Filtered - filters all spots manually marked to be filtered in Spot View or Replicate View.

All filters have a ▶ **Filter** button. Press this button to see how many spots that will be filtered by this filter.

Normalization

J-Express Pro includes three normalization methods named **MPI, Median** and **Lowess**. All methods require a two-channel dataset ordered in a <ch1 ch2 ch1 ch2 etc> format. If

the data is not organized this way, you can do so by either creating a script that reorganizes the columns in your dataset, or manually slide the columns in the tabular view and define a new dataset. The columns can be slid by clicking and dragging the grey area above the columns.

- Normalization can be carried out on the entire array, a block or a group of blocks of the array. The groups are defined during the Quality Control.

- Median - this is a type of single-parameter linear normalizaion. It normalizes the data so that the median intensity is the same across the entire array.

- MPI - Martin Vingron at the Max Planck Insitute (MPI) in Berlin has contributed the MPI normalization. It uses a regression method and applies a transformation of the channels so that the ratio of most (including first those with high intensity) spots becomes 1. For method description see

> **Processing and quality control of DNA array hybridization data**
> Beissbarth T, Fellenberg K, Brors B, Arribas-Prat R, Boer JM, Hauser NC, Scheideler M,Hoheisel JD, Schuetz G, Poustka A, Vingron M
> Bioinformatics; 11.2000; 16(11): 1014-1022.

Lowess - normalizes intensity dependent effects of the data, particularly at low and high intensities. These effects may cause a "banana" shape of the data, which cannot be corrected by linear normalization. Lowess combines features of linear least squares regression with features of non-linear regression, by fitting simple models to localized windows of the data to build up a function that describes the deterministic part of the variation in the data, point by point.

The Lowess procedure is described in the article Cleveland, W.S. and Devlin, S.J. (1988) "Locally Weighted Regression: An Approach to Regression Analysis by Local Fitting," *Journal of the American Statistical Association*, Vol. 83, p. 596-610.

When clicking any of the types of normalization, a window showing the before and after normalization is opened.
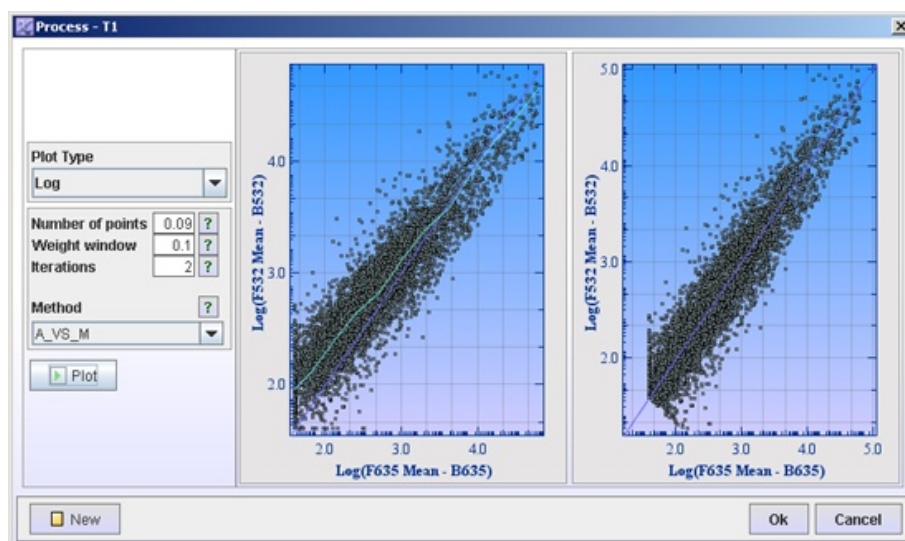
Set the Plot Type and other parameters. Click on the question mark behind the parameters to get information on the particular parameters. Click the ▶ Plot button to see the before and after normalization plots. Right click on a plot to change its appearance. For further information on customizing plot appearance see section 3.8.2

You can define which genes you want to use as a reference for the normalization (such as control genes) by clicking the Normalization source button. From the window that opens you can create a sub-filter that removes all genes but the ones you want to use as a basis for normalization.

One-channel data can also be normalized trough a script. A script to do just this can be found in the resources/scripts folder. In the one channel case, all columns will be normalized with regards to the first column.

You can remove a certain percentage of a quantile by entering the desired value in the **Subtract an X% quantile** box. To end the refinement process after normalization, click **OK**, or click the **>>** button to continue to the last step.

If the Lowess normalization method is selected a **Parameters** button appears in the lower right corner of the window. Click this button to set the parameters used by the Lowess method. In the Lowess parameters window that appears, you can click the question mark to get a short explanation of each parameter. The parameters are:

- Number of points – sets the number of points used for the regression line. Enter a new value in this box if needed.

- Weight window – sets the width of the Lowess window. Enter a new value in this box if needed.

- Iterations – this parameter sets the amount of Lowess iterations to use. Enter a new value in this box if needed.

- Method – This parameter defines the type of plot to base the Lowess regression line on. Select a new method from this pull down menu if needed.

The final step of the raw data refinement is to choose which transformation method to use. Select the method from the list provided.

- Log ratios – transforms the data by the logarithm of the ratios of the channels

- Ratios only – transforms the data into the ratios of the channels

- No Ratios – leaves the data as it is

Click the **OK** button to complete the refining process. A new data node will be created in the Project Tree below the raw data node.

Other

- Plot - if you want to see a plot of the your data after having done some filtering or normalizations, add a Plot process and move it to the position right after the processes you want to see the result of. Click in the run column of the plot process. This performs the above processes and plots the graph result. You can set different colors for filtered and non-filtered spots. You can also choose whether you only want to plot the filtered or non-filtered or both by checking the check-boxes.

- Value Boundary - Set all fields with value greater than, equal to or less than a certain value to a specified value. This can for instance be used to setting a floor value for very low intensities. Use the target button and filter the attributes you want to keep as they are, without being replaced by a floor value.

- Spike Viewer - Spike Viewer is used to examine the controls printed on the arrays.
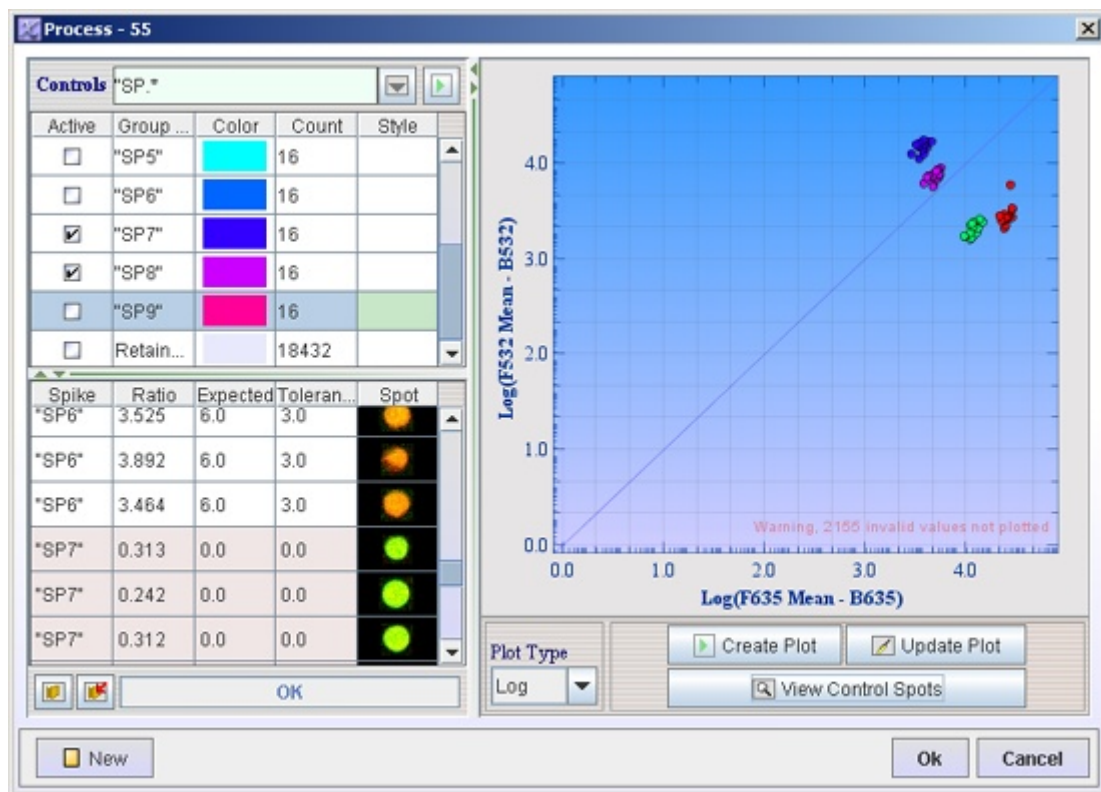
On the left of the divider there is a search field and two tables. **Locate the controls** by typing a regular expression in the text field behind the label Controls and press ▶. The 10 last used search phrases are saved and can be selected by clicking the ▼ button. The search result will be displayed in the top table. All spots from one control that are printed on different places around the array, make up one Group. The number of members to a group is displayed in the Count column. Each group get its own color. You can change these colors by clicking on the colored rectangles and choosing a different color.

Choose the **Plot Type** and press ▶ **Create Plot**. The spikes checked in the Active column will be plotted with their specific colors, in the graph display window. Since you know what the ratio for the controls should be, the plot lets you see if the data are skewed in any direction. If you now look at the bottom table, you can see each of the control spots (or spikes) listed.

Click **Copy Controls to Registry ( )**, then click the **Open Spike Control Registry ( )** button. Here you can set the **expected ratios** and **tolerance limits** (Setting the value 1 means tolerance of +-1). Click ok. Look back at the bottom table. All spikes that have ratios within their expected ratio + tolerance limit will have a white row, while the others will have rows that are colored red.

Click the **View Control Spots** button. This will import all the control spots from the array and display them in the Spots column of the lower table. You can now examine the spots to see if the same control looks the same across the array. This may also help to explain the reason if any spikes have ratios outside the limits. You can add other controls that have a different regular expression by locating them the same way as before.

Click **Copy Controls to Registry ( ).** This will add the new controls to the registry. To plot the new controls, you only need to **Update Plot**. It should only be necessary to press the Create Plot button the first time, or every time you change the Plot Type.

- Spot Image View - This component is similar to View Combined Image (section GenePix), with the difference that it lets you see which spots have been filtrated during the processing.

- Replicate View - Replicate image view can be used to examine replicate spots on an array. In the table that opens all unique IDs will be listed, whether it has been filtrated through filtering methods or manually filtering, number of replicates on the array, and some ratio statistics. Select a row and click the **Details Selected** button to get details on each of the replicate spots. You can also filter spots from here if you wish. Click ok to add Replicate View to the Processing Batch.

Click **Ok** to add a process to the Processing Batch, **New** if you change your mind and want to go back to the window where you select the processes, and **Cancel** to go back to Processing Batch without adding anything.
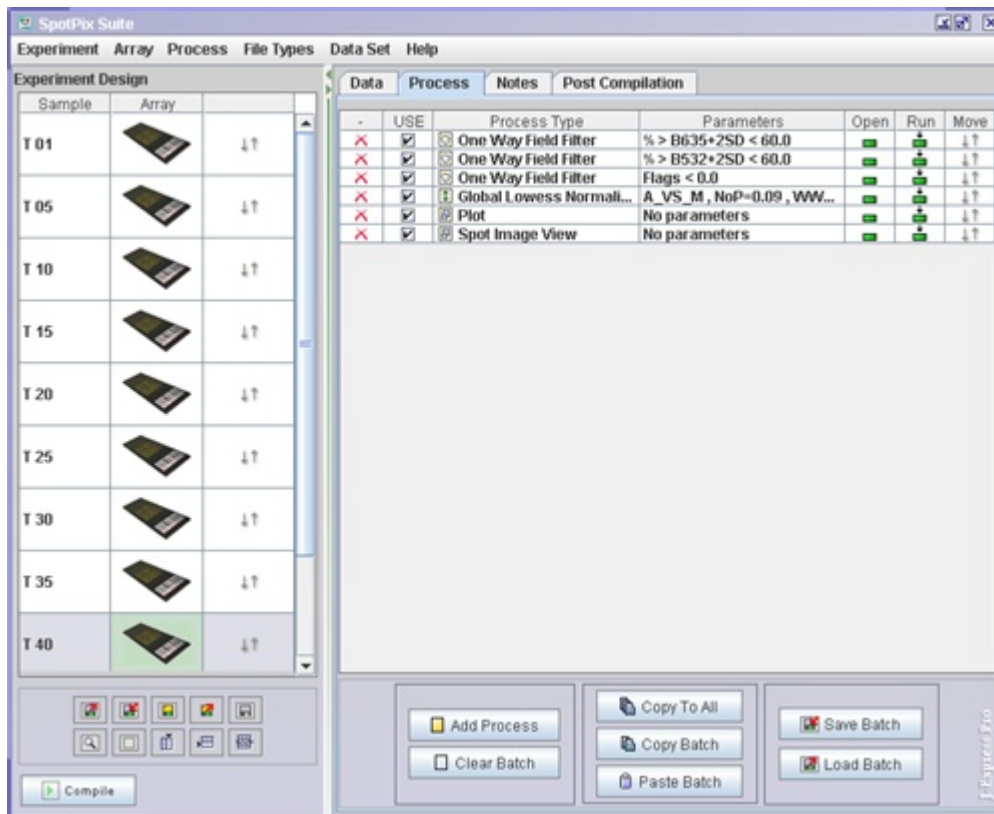
The processes listed in the Process Batch will take affect first when you press the ▶ Compile button, which is located in the lower left hand corner of the SpotPix Suite window. This means that you can play around with different processes and see the effect of doing different types of filtering and normalizations, before the final dataset is created and added to the J-Express Pro project tree.

To run the processes, click one of the rows in the **Run (▲)** column. This will process all the processes from the top of the list, down to the row you clicked. If you wish to change the order of which the processes are carried out, move a row by clicking and dragging in the **Move (↓↑)** column. If you want to reopen any of the processes click in the **Open (▭))** column of the row of the process you want to open.

If you want the same processes to be carried out on all of your arrays, click the ▤ **Copy To All** button.

3.1.6 GenePix
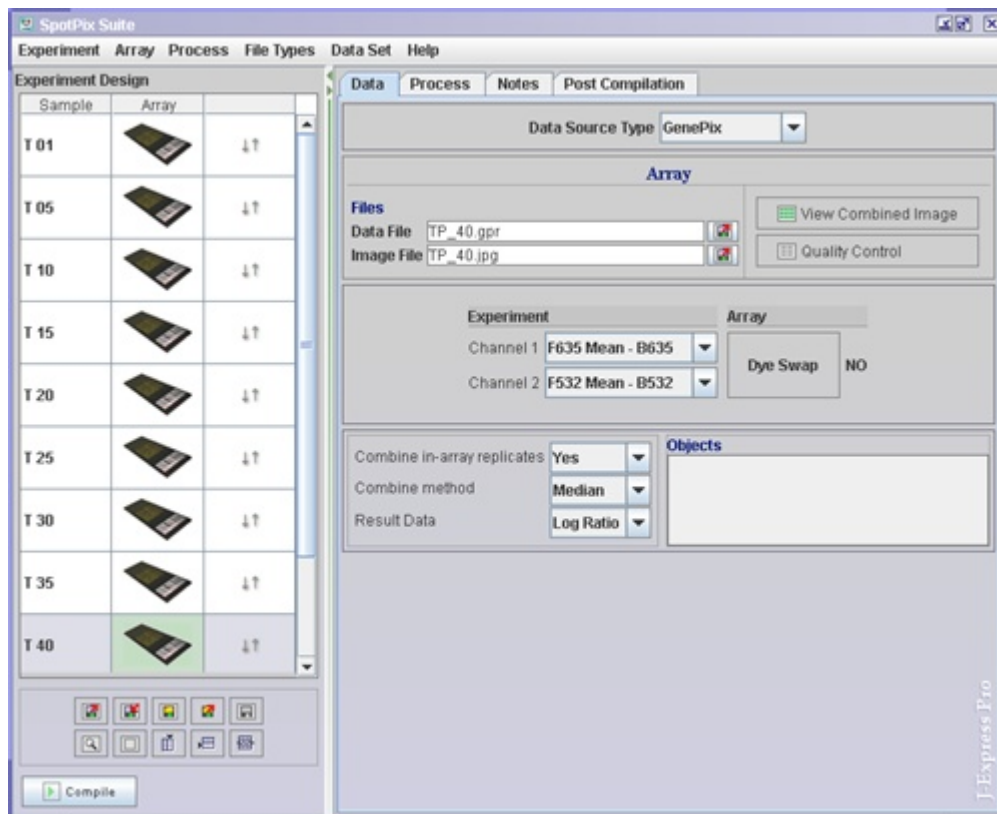
To link the genepix files to the **array images (****)**, click on each array and set the genepix data and image files by clicking the **Load Experiment(** **)** buttons in the Data Tab. It is now a good idea to save the experiment. Save by clicking the **Save Experiment(** **)** button, to the left of the divider, underneath the experimental design.

Experiment | Search for image files in folder

Search for image files in folder is only available for GenePix. The .gpr files are searched for the name of the image files. Next the images in the selected folder are mapped to the .gpr file. This is valuable if a project file is sent to someone else who already have the image files. This basically sets the correct file path.

Experiment/Array (Data Tab)

Set the preferred selection for **Channel 1** and **Channel 2**. For instance, if Channel 1 is set to F635 Mean - B635, this means that the color of this channel is red (wavelength 635 nm) and that mean pixel intensity is used for the foreground. (Green light has wavelength of 532 nm.)

- F – foreground
- B – background
- 635 - wavelength of red light
- 532 - wavelength of green light

If dye swap has been carried out on an experiment/array, J-Express Pro needs to know this. If that is the case, click the **Dye Swap** button on the dye swap array.

Experiment

Select the preferred values for the combo boxes at **Combine in-array replicates**, **combine method**, and **Result Data**. Combine in-array replicates means that replicates on the same array will be combined in some way, so that they are all represented by just one value. If the Combine in-array replicates is set to yes, remember to also set which method to used to combine the replicates.

Certain **Objects** can be saved to a project. An Icon will appear in the Object field. To view or continue working with an object, double-click the icons. There are two different types of object that can be saved to a project:

- **Spot View**   and **Selection Container**

These are described in more detail in section .

Quality Control

The **Quality Control** button opens a window that allows you to examine the quality of your chip. This may help you decide what how to normalize your data.
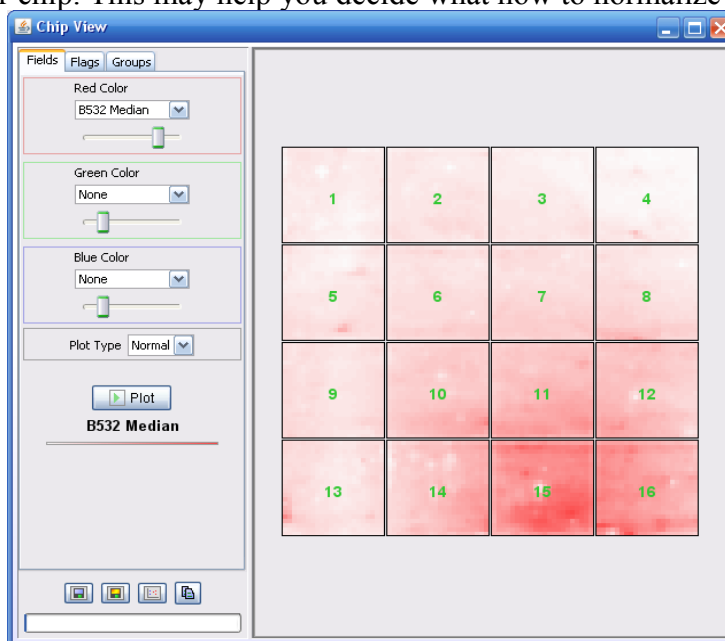
The Chip View window displays an array to the right, and three tabs to the left. The three tabs are **Fields**, **Flags** and **Group**s.

Fields

The fields tab contains a **red**, a **green** and a **blue channel.** The three channels represent the primary RGB colors. Each of these has a selection of settings that can be

chosen from the combo boxes. The various settings allow you to examine how the background intensities are in comparison to the foreground intensities for different areas of the chip. Play around with different selections in the combo boxes. Slide the color bars to tune the color intensities. Press the **Plot** button to update the chip view. Looking at the picture above, which plots the background distribution in red and blue channel, it is apparent that the background intensities are not the same all over the chip. It is also possible to view the real chip image by **right-clicking** any of the blocks in the array and select **View Chip Image**.

The Chip View can be saved as an image by clicking the **Save Array View Image(⬜)** button. To save the scale bars for the different channels, click the **Save Array View Scale(🟨)** button.

Flags

The flags tab allows you to see if there are many spots not found by GenepPix. Click **Add**. Next click the 0 in the new added line, and choose -50. You can change color by clicking on the black rectangle. Click **plot**. The chip view will now show the Spots not found by GenePix. -100 means spot missing.

The Chip View can be saved as an image by clicking the **Save Array View Image(⬜)** button. To save the **Flag** and **flag color** as an image, click the **Save Array View Scale(🟨)** button.

Groups

You can divide the blocks in the array into different groups. Select the blocks you want in a separate group by **clicking and dragging** the mouse over them (the entire block has to be inside the square that is drawn when clicking and dragging, in order to be selected). **Right-click** on one of the selected blocks, and select **Create new Group**. The selected blocks will now be removed from the original group and added to the new group. If you want to add some blocks to a group that already exists, select the group you want it added to in the group list, click and drag mouse to select the new block(s), right-click and select **Add Selection To Selected Group**.

The selected blocks can be given their own color by **Right-clicking** on one of the selected blocks, and selecting **Set Block Color**.

It is also possible to plot the various fields available against each other for a group. Right-click and select **Plot Block**, or press the **Plot Selected Blocks(⬜)** button at the bottom left hand corner or the Chip View window. This will open an **array scatter plot** window.
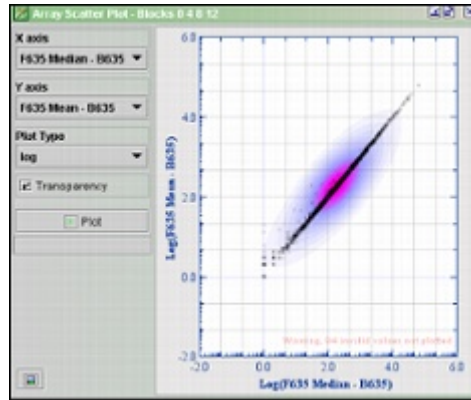
The plot above plots the median/mean ratio for blocks 0, 4, 8 and12

Select the fields you want to plot at the **X-axis** and **Y-axis** pull-down menus. Choose a **Plot Type** and **click Plot**. The plot can be saved by clicking the **Save Array Plot Image (🖫).**

View Combined Image

The 🎛 **View Combined Image** button opens the **Chip Image View** window. Note that this only works for gene pix arrays. Also, it only works if you have added the jpg-images for the gpr files.
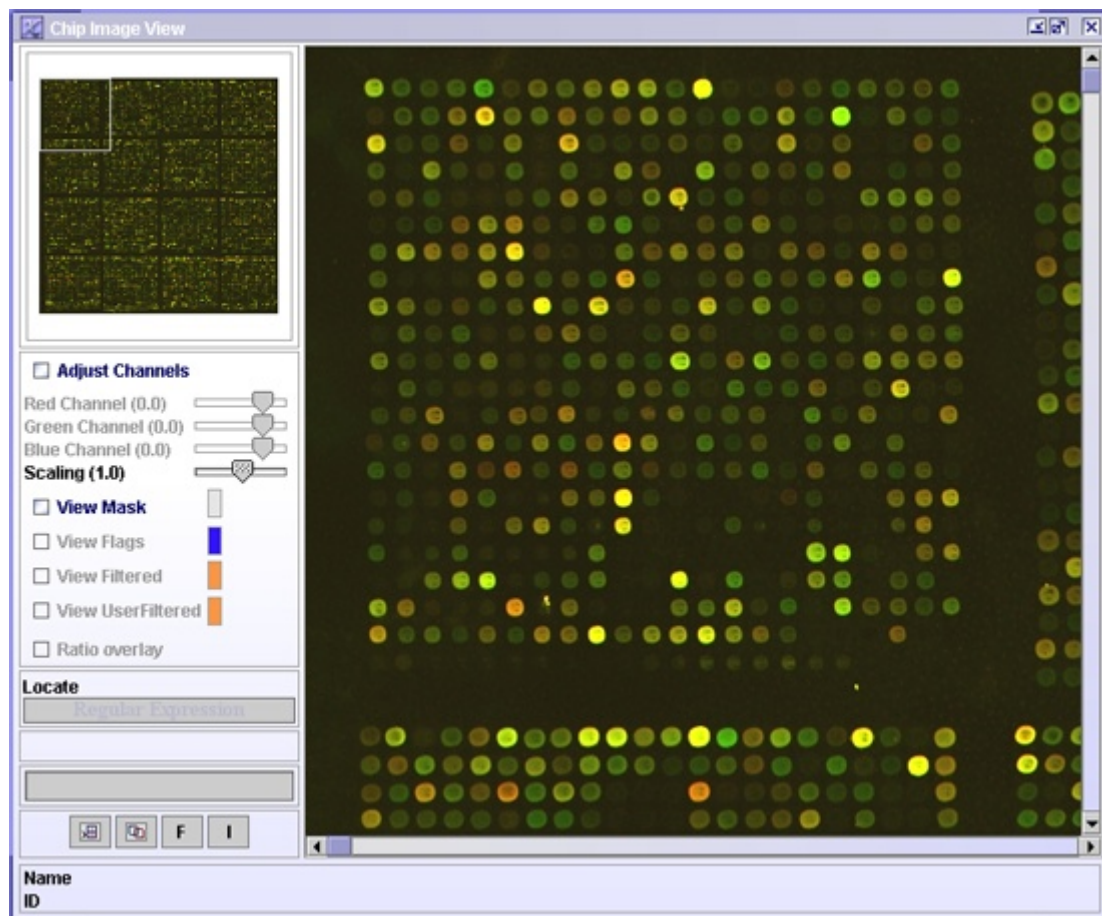
The picture to the right depicts the microarray combined of the scanned pictures from the red and green channels.

- **Adjust Channels** - check to adjust the RGB color contrasts.
- **Scaling** - slide to zoom in or out of the picture

The color of the next four check buttons can be changed by clicking on the colored rectangle.
- **View Mask** - check to draw a circle around the spot. This tells J-Express where the spots are, and can also make it easier to see the spots.
- **View Flags** - check to draw a circle around the flagged spots
- **View Filtered** - check to draw a circle around the filtered spots. See that no "good" spots are filtered.
- **View User Filtered** - By clicking the F button you can manually filter spots. You can also manually filter spots in the replicate view window. When selecting this checkbox, you can color the frame of manually filtered spots.
- **Locate** - Type the name or id of a spot to have it highlighted.

Open Linked File Value Table (⊞):
Click Open Linked File Value Table (⊞) button to open a spreadsheet containing all the raw data values associated with each spot on the array.

Link Events To Open Value Table (⊡):
Linking events to open value tables means that file value tables that are open, will be linked to the spots in the picture. When clicking on the spots, the corresponding entry in all the file value tables will be highlighted. This way you can see what values the spot you click has in the data file. **Note: The "View mask" checkbox has to be selected for this to work.**

F:
If you want to filter the spots manually, you can do so by click the button labeled **F** before clicking spots you wish flag.

I:
To examine some spots further, make sure that "View Mask" is checked and click the button labeled **I**. This will open an empty Selection container. Click on spots you want to add to the Selection container. If more than one selection container is open simultaneously, the spots that are clicked will be added to all of the selection containers that are open.

The Selection Container

The selection container contains location data and id of the spots you clicked. Selecting entries in the Selection Container will mark the corresponding spots with a light blue square in the Chip Image View window. If a **File Value Table (** **)** is open, the selection should be marked there as well. From the File Value Table you can see the values of the entries in the Selection Container.

To clear a selection container press the **Clear Selection Table (** **)** button. To delete rows from the selection container, select the rows you want to delete and press the **Remove Selected Elements (** **)** button.

Find in-Array Replicates (**):**

To locate in-array replicates to the entries in your selection container, press the **Find in-Array Replicates (** **)** button. This will look up all the selected entries in the selection container and add the replicates to the selection container if they exist.

Storing the selection container in a project:
The selection container can be saved to a project. To do so press the **Store in**

Experiment (**)** button. The icon will appear in the **Object field** of the data tab. To start a new selection container, press the **New Table** button.

Click **Get Spot Images (** **)** button to view the selected spots on all the different arrays.

The SpotView can be **saved (** **), printed (** **), exported to HTML (** **)** and **stored in an experiment (** **).** If the spotview is stored in an experiment, the icon

 will appear in the **Object field** of the data tab. The image can also be copied to clipboard by pressing the **Copy Image to Clipboard ( )** button.

3.1.7 Affymetrix
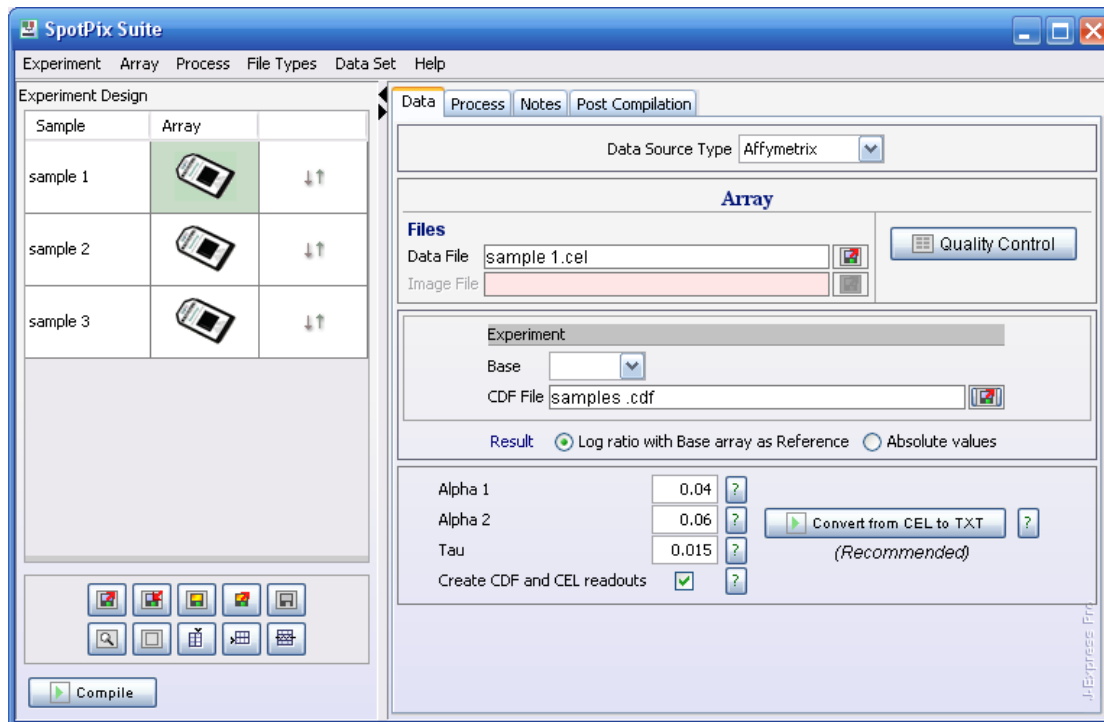
To link the affymetrix files to the **array images ( )**, click on each array and set the affymetrix data and image files by clicking the **Load Experiment( )** buttons in the **Data Tab**. Set the **CDF File.** If you need to do normalizations or present your results as log ratios, select one of your arrays to use as **Base**.

Decide whether you want your result presented as **Log ratio** or **Absolute values.**

It is now a good idea to save the experiment. Save by clicking the **Save Experiment( )** button, to the left of the divider, underneath the experimental design.

Quality Control

The **Quality Control** button opens a window that allows you to examine the quality of your chip. All statistics used on probe pairs and probe sets are taken from http://www.affymetrix.com/support/technical/whitepapers/sadd_whitepaper.pdf
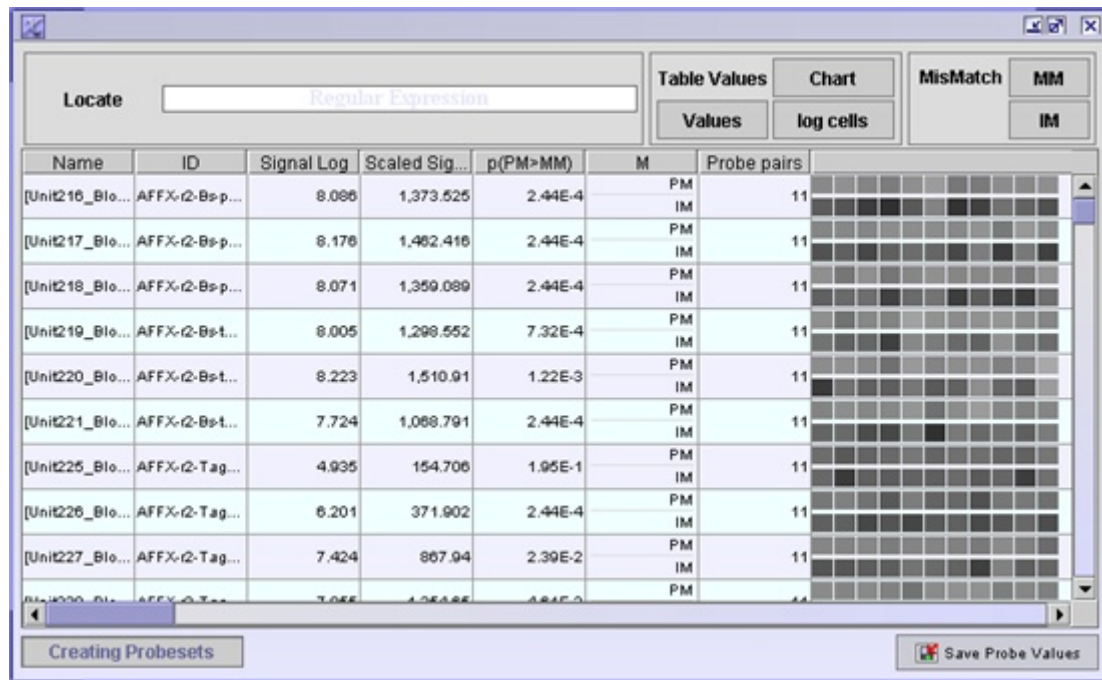
The table shown in this window displays information on each probe pair in your dataset. A probe pair consists of two probes, one that **matches perfectly** to the target mRNA (**PM**) and a probe where one **mismatch (MM)** has been introduced.

Each row has information on a probe set. A probe set is a collection of probe pairs, all with the same mRNA as a target, located at different places around the array. Ideally all PM probes within one probe set and all MM probes should have the same intensity, or at least similar ratios of each probe pair. This is often not the case.

The table values can be presented in three different ways:
- **Chart** - each row has a graph area, where a red dash is drawn for each PM and a blue dash for each MM. High intensities are drawn near the ceiling of the row, low intensities near the floor of the row.

- **Values** - the absolute intensity values

- **Log cells** - displays the affymetrix scanned cells. These are in a shade of grey. A black square means no intensity, a white square means maximum intensity.

The probe that has a mismatch introduced, should have lower intensity than its PM variant. If the MM probe has higher intensity than the PM probe, an **ideal mismatch (IM)** is calculated. To view the calculated ideal mismatches click the IM button. Click forth and back between MM and IM buttons to see where an ideal mismatch has been calculated.
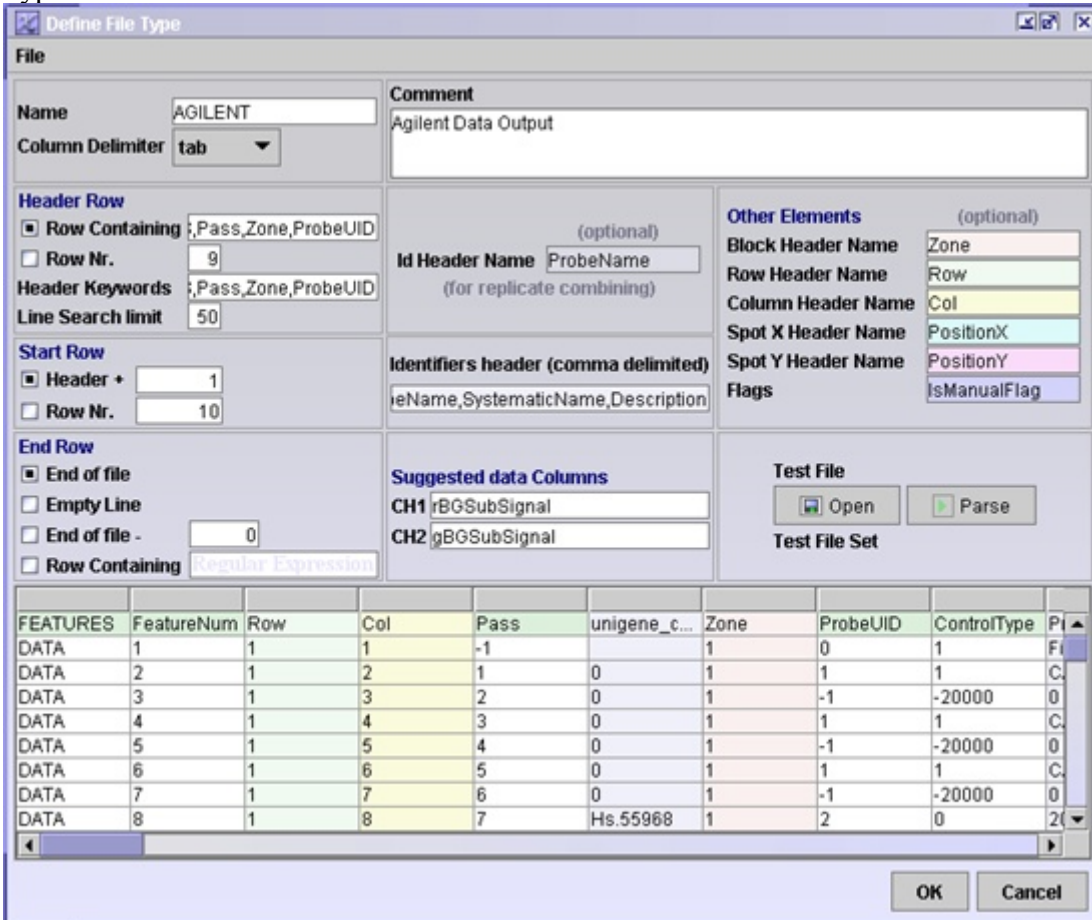
If you are looking for particular probes, you can search for them using a regular expression in the search field provided.

3.1.8 Tabular

If you have a data file type that is not supported here, you can define your own file type for importing data. All user defined file types will be available from the File Type selection combo box. **Select a file type** from the selection combo box, or click the **File Define Menu** button to define a **new** file type, **modify** an existing file type or **import** a user defined file type.

The **File Define Menu** holds a list of all User Defined File Types. From here you can **define new** file types**, edit** a selected file type from the list or **delete** a selected file type from the list. It is also possible to **Import** file types defined by others or **Export** the defined file types so that it can be sent to other people. When you have finished editing your File Define Menu, click OK to go back to the SpotPix Suite.

Click the ⊞ **Create New** button to define a new file type to open the Define File Type window.

Fill in the **File Type Properties** to define the file type.

- **Name:** the name will be added as an identifier of this file type to the File Type combo box in the Data Tab in SpotPix Suite.
- **Column Delimiter:** The columns can be delimited by either **tabular**, **comma** or **space**.
- **Header Row :**
  - **Row Containing:** Type the headers for this filetype, separated by commas.

- o **Row Nr.:** If you know the header row to always start on the same row number, you can select this option and enter therow number.
- o **Header Keywords:** If you drag and drop files into the SpotPix Suite Experiment Design area when setting up the experiment, the header keywords typed in this text field will be used to identify the file type.
- o **Line Search limit:** Normally the header row is found somewhere near the top of a file. Some files can be quite large, and searching through the entire files for the header keywords can be time consuming. It is therefore a good idea to set a limit for how many lines of a file should be searched. If no hit on the header keywords are found within the first e.g. 50 lines, the file will be reported as unknown file format.
- **Start Row:** The index number of the first data row
  - o **Header +:** use this if you do not know the exact row number, but you know that the first row starts a certain number of rows after the header row.
  - o **Row Nr:** If you know the first data row to always start on the same row number, you can select this option and enter the row number.
- **End Row**
  - o **End of file:** end row is the last row in the file.
  - o **Empty Line:** end row is the last row before an empty line.
  - o **End of file - :** end row is a certain number of rows before end of file, i.e.: end of file **minus** a certain number of rows.
  - o **Row Containing:** end row is a row containing a specific text or a regular expression.
- **Comment:** short description of this file type.
- **Id Header Name (optional):** the name of the column containing the spot identifications
- **Identifiers headers (comma delimited):** type the header of the identifier columns in this text field, and separate each header by a comma.
- **Suggested Data Columns:** here you can set default columns to use for a particular filetype. When a file is dropped into the SpotPix Suite Experiment Design area when setting up the experiment, the default suggested data columns will automatically be set for channel 1 and channel 2.
- **Other Elements (optional)**
  - o **Block Header Name:** this column holds which array block a spot belongs to.
  - o **Row Header Name:** this column holds which array rows a spot belongs to.
  - o **Column Header Name:** this column holds which array column a spot belongs to.
  - o **Spot X Header Name:** this column holds the x coordinates of the spots. This will be used by the Quality Control component to identify where the spots are.
  - o **Spot Y Header Name:** this column holds the y coordinates of the spots. This will be used by the Quality Control component to identify where the spots are.
  - o **Flags:** this contains information on whether the spots are flagged or not.

You should now test your file format. Click the ▣ **Open (Choose a test file)** button. See that the **Test File Not Set** label changes to **Test File Set**, and click the ▶ **Parse** button. The result will be displayed in the lower part of the window. If everything looks as you expected, click **OK**. If it doesn't look as it should, check that all your parameters are entered correctly. Remember that all Header Names must be entered exactly as they appear in the file.

Experiment/Array (Data Tab SpotPix Suite)

Except for a couple of things, the rest of the experiment setup for user defined data type is very much like GenePix data type, see section 3.1.5.

If your dataset have only one channel, check the **Single Channel** check box, and set which array you want to use as a **base sample** for log ratios and normalizations. Set the preferred selection for **Channel 1**.

If your dataset have **two channels**, see that the Single Channel check box is unchecked. Set the preferred selection for **Channel 1** and **Channel 2.**

Note: No objects can be saved to the Project Dataset data source type.

Quality Control

The Quality Control of User Defined data type is the same as for GenePix, see section 3.1.5.

See section 3.1.4 for information on filtering and normalization of the data.

3.1.9 Project Dataset

Project Dataset is used to refine raw data that has been loaded as tabular data (section 3.1.2). Such data has every other data column for one channel and the rest of the data columns for the other channel. Select the raw data node in the J-Express Pro project tree and open the SpotPix Suite by clicking the **Open Spot Pix Suite (▦)** button, or selecting **Raw Data | Open Spot Pix Suite.**

Set **Data Source Type** to **Project Dataset**. Click ▦ **Create Experiment**. This will set up the entire experiment for you, selecting the first and second data column as the two channels of array one, third and forth data column as the two channels of array two, etc. You can change the array data columns for an array by selecting an array from the Array column in the Experimental Design table. Then set the columns you want to use as the **Foreground** and **Reference columns** in the Experiment / Array Specific section. Set the column containing **Replicate ID info** from the combo box, if it exists.

It is also possible to set up the experiment manually as described in spotpix, section 3.1.3. If you choose to do so, you have to click on each array in the Array column, and set the **Foreground** and **Reference Columns**.

Experiment

Select the preferred values for the combo boxes at **Combine in-array replicates**, **combine method**, and **Result Data**. Combine in-array replicates means that replicates on the same array will be combined in some way, so that they are all represented by just one value. If the Combine in-array replicates is set to yes, remember to also set which method to used to combine the replicates.

Note: No objects can be saved to the Project Dataset data source type.

See Processing/Refining Raw Data, section 3.1.4 for information on filtering and normalization of the data.

3.2 Robust Multi-array Average (RMA)

RMA is an algorithm used to create an expression matrix from Affymetrix data. The raw intensity values are background corrected, log2 transformed and then quantile normalized. Next a linear model is fit to the normalized data to obtain an expression measure for each probe set on each array. RMA is very easy to use in J-Express, a description follows further down on this page. All transformations and normalizations change the data, and it is highly recommended to understand how. For more on RMA, see:

- *Irizarry, R. A., Hobbs, B., Collin, F., Beazer-Barclay, Y. D., Antonellis, K. J., Scherf, U.,*
  Speed, T. P. (2003). Exploration, Normalization, and Summaries of High Density
  Oligonucleotide Array Probe Level Data. Accepted for publication in Biostatistics.
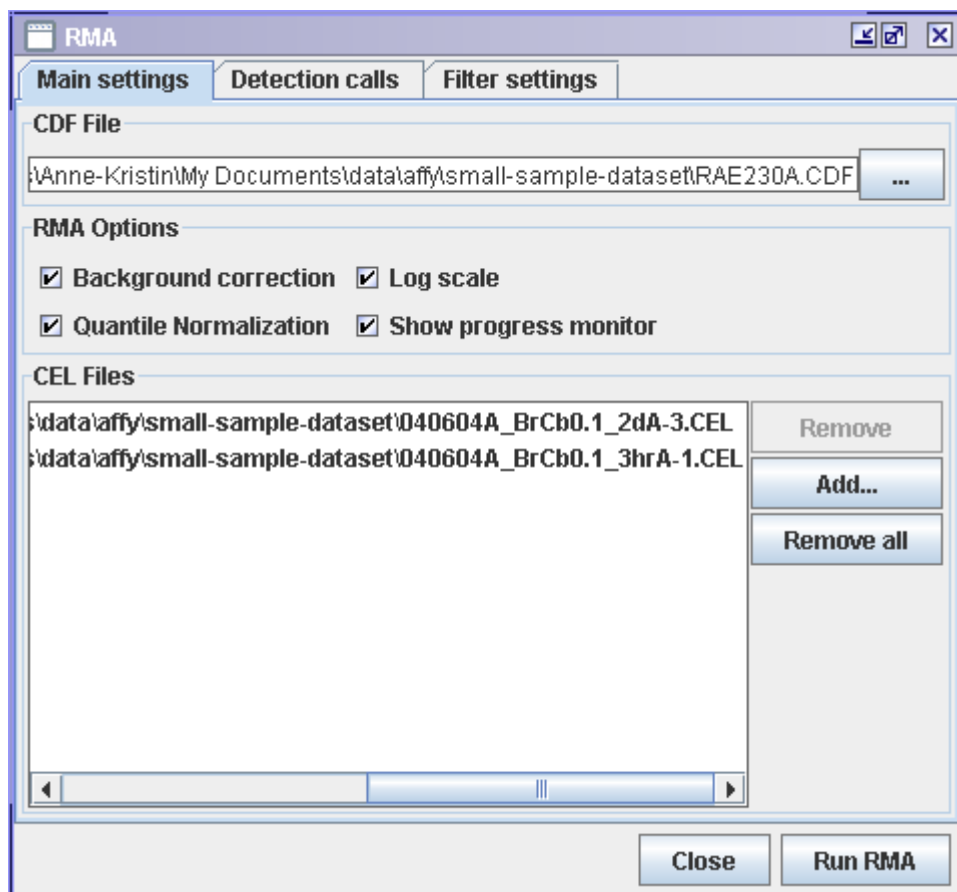
3.2.1 Memory usage:

RMA uses a lot of memory since it works on all arrays at the same time. If you encounter memory problems, you can increase the java heap size. See here how you do that for J-Express.

3.2.2 How to load affymetrix data using RMA:

1. Open RMA from **File | Load Affymetrix Data Using RMA**
2. In the **Main settings** tab:
    1. Locate the **CDF File**. J-Express will automatically look for CEL files in the same folder as the .CDF file and add them to the **CEL Files** list.
    2. If .CEL files have been added to the **CEL Files** list, view the list to see that it contains the ones you want to use. If you need to make any changes to it, use the **Remove** and **Add** buttons to get the files you want.
    3. Select the **RMA Options** you want to use. Large datasets, e.i. more than 10 .CEL files may take long to process. It is then a good idea to check the **Show progress monitor** so that you can see that it is working and to monitor memory usage.
3. I the **Detection calls** tab:
    1. The detection algorithm calculates a score for each probeset that is used to call a transcript present, marginal or absent. The sensitivity and spesificity of the detection algorithm can be adjusted by changing the Alpha and Tau parameters.
        1. **Tau** influences the detection p-value, which is used to call a probepair present or absent. Increasing the threshold Tau can reduce the number of false Present calls, but may also reduce the number of true Present calls.
        2. Probe sets with detection p-value lower than **Alpha 1** are called **Present**.
        3. Probe sets with detection p-value higher than **Alpha 2** are called **Absent**.
        4. Probe sets with detection p-value in-between **Alpha 1** and **Alpha 2** are called **Marginal.**
4. In the **Filter settings** tab:
    1. **Minimum percentage of 'Present' genes:** means that genes that have less than e.g. 50 % Present calls for all arrays will be removed.
    2. **Maximum percentage of 'Absent' genes:** means that genes that have more than e.g. 50 % Absent calls for all arrays will be removed.

Press **Run RMA**.

3.3 The Project Workspace

The project workspace is organized around the project tree. The project tree is rooted in the project folder, and provides easy access to any subsets of the data that you define by branching. A module is a subset of data in J-Express Pro. A new branch is created every time you add a data file, clone or transpose a dataset, or by branching off a selection of profiles. Whenever a set of profiles is branched, a new branch is added to the project tree under the currently active node. Data analysis can be performed on any node in the project tree below the project folder. An exception is for raw data sets that need to be refined to gene expression data sets before analysis. Any node in the project tree can be renamed by double-clicking its label, and then entering a new label and pressing enter. The number of rows and columns of data in the dataset is shown below the project tree. The image below shows a project with 121 profiles with 18 states for each profile.
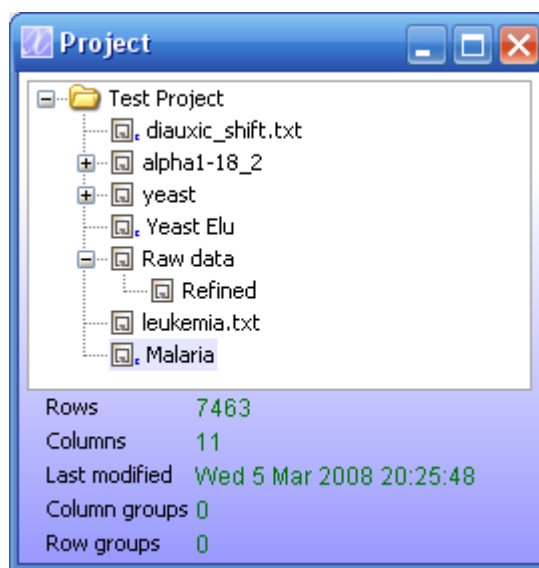
A project tree with branched subsets.

In addition to simple branching of data subsets, J-Express Pro supports the following basic operations on data nodes in the project tree:

1. Clone

2. Clone to root

3. Transpose

4. Delete

Advanced operations on datasets such as filtering and creating a sub data set are described in separate sections in this chapter.

To clone a node in a project:

Select the node you want to clone by clicking it in the Project Tree. On the J-Express Pro menu bar select **Data Set | Clone Dataset**. A new node containing a copy of the selected dataset is created on the same level in the Project Tree.

To clone a node to the root of the Project Tree:

Select the node you want to clone to the root of the Project Tree by clicking the node in the Project Tree. On the J-Express Pro menu bar select **Data Set | Clone Dataset to Root**. A new node containing a copy of the selected dataset is created on the top level of the Project Tree.

To Transpose a node in a project:

Select the node you want to transpose by clicking on it in the Project Tree. On the J-Express Pro menu bar select **Data Set | Transpose Data**. A new node is created on the same level in the project tree containing a transposed version of the data. Transposing the data is handled similarly to a matrix transpose operation in linear algebra, an MxN matrix is turned into an NxM matrix by letting the rows in the original data set become columns in the transposed dataset.

To delete a node from a project:

Select the node you want to remove from the project by clicking it in the Project Tree. Right-click the dataset and select delete dataset.

All nodes are labeled with an icon related to the method used when the set was created.

Note: Selecting a different node in the Project Tree simply selects the dataset corresponding to that node, and does not update any method windows you may have open with the new data.

Project tree icons		The following icons are symbols for **result windows** (generated by **put in tree** 🔲) and can be opened by a double-click
📁	Project Folder (top level)	
🔲	Data from file	
🔲	Transposed data	🔲 Dendrogram
🔲	Hierarchical Clustering data	🔲 PCA Plot
🔲	K-Means Clustering data	🔲 Gene Graph
🔲	PCA data	🔲 Thumbnails
🔲	Self Organizing Map data	🔲 MultiDimensional Scaling
🔲	Filtered data	🔲 Spot View
🔲	Cloned (or other subset generating method)	🔲 GSEA result
🔲	Profiler	🔲 SAM result
🔲	Extracted rows or columns	🔲 Rank result
		🔲 A cluster list

Similarity Search	
Group controller	
High level normalized data (from the create sub dataset component)	
The columns of this dataset has been re-arranged	
Feature Subset Selection	
Multidimensional Scaling	
Impute Missing Values	
Search and Sort Search and Sort	
Data from Corresponence analysis	
Data from fold change	
Data from gene graph	
Data from gene ontology	
Data from gene set enrichment analysis	
Data from pathway analysis	
Data from gene graph	
Data from gene graph	

Dataset Viewer

Right click a node in the project tree and select **View dataset** to display basic statistics for the node, and choose which info columns to keep visible. Alternately you can click the button on the J-Express Pro tool bar. From this component, you can change all elements in the dataset main matrix. The Values tab shows a spreadsheet of the dataset. Double-click a cell in the spreadsheet to alter its value. Note that this change will take effect immediately. Right-clicking an info row or column will also give you a choice to delete that row/column. New rows or columns can also be added from this menu. To keep track of the changes you've made, check the **Submit changes to meta data** box.

The **Value Distribution tab** shows a histogram of the distribution of the values in the dataset. In addition, the mean, maximum/minimum values and the median of the dataset is shown, as well as the number of missing values replaced/interpolated. The number of Histogram Bins refers to how many bars should be used to represent the value distribution.

The **Info Fields tab** lets you select which fields should be shown when displaying additional information on a profile, both for rows and columns. Check the items you wish to display, uncheck to stop displaying them. Check the items you wish to display, uncheck to stop displaying them.

3.3.1 Project Thumbnails and Info/Metadata

Below the Project Tree window are two windows displaying information about the currently selected dataset at a glance. Project thumbnails give you a low-detail graph showing all the profiles in the dataset (this can be changed to only show the selected profiles).

Click the two squares at the lower right of the window to toggle between showing the current selection, or all profiles.
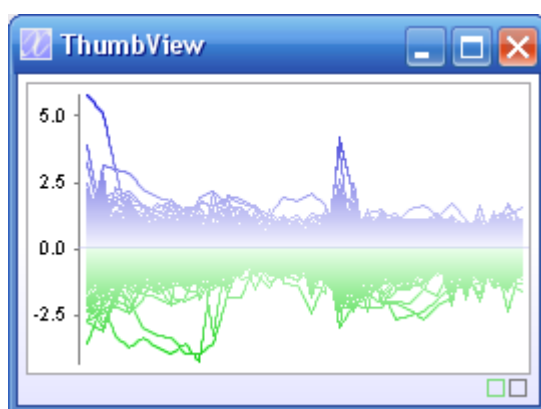
A project thumbnail showing a dataset containing 2467 profiles.

To change the colors in this window, go to settings -> options and change the global color scheme.

Meta Info Icons

 Root, This node does not represent a analysis step

 Appears for projects created by J-Express Pro, version 1.0. These projects have do not have a Parameter Tag, and is therefore given the tag Old Meta Tag

 A DataSet produced by the SpotPix suite. This is loaded from raw data.

 A Sample in the SpotPix experiment

 An actual array in a sample in the SpotPix experiment

 A process bach in an array

 The dataset has been unlinked from the parent dataset. See explanation above (link)

 The columns of this dataset has been re-arranged

 The dataset has been created by the similarity search component

 The dataset has been created by the profile search component

 The dataset has been created by the create groups component

 The dataset has been created by the group manager component

	A process in the process batch
	A loaded dataset (tab delimited)
	A cluster in a k-means clustering window
	A cluster in a Self-organizing map component
	Data produced by a hierarchical clustering branch
	Data produced by a GSEA branch
	Data produced by a SAM branch
	Data produced by a hierarchical clustering branch
	Data extracted from a PCA window
	The dataset has been linked to parent and does no longer contain its own data. The data in this dataset belongs to the parent and changes made before this step on this dataset will be discarded. DataSets are linked to save space. After for instance a filtering step, only a few data rows are removed and it is a waste of space to copy all rows in the new dataset. The Link and Unlink steps are normally transparent from the user and handled automatically by J-Express. It is however possible for the user to relink or unlink a dataset manually

	The dataset has been created by the filter dataset component
	The dataset has been created by the create sub dataset component
	The dataset has been created by the feature subset selection component
	The dataset has been created by the impute missing values component
	The dataset has been created by the multidimensional scaling component
	The dataset has been created by the correspondence analysis component
	The dataset has been created by the dataset viewer component
	The dataset has been created by the search and sort component

The **User Info tab** provides a text area where notes can be entered without leaving J-Express Pro. These notes will be saved with the project. Note: Each dataset in a project has a separate space available for notes. Thus, you can have one set of notes describing the entire dataset, and another describing a particular subset of interest. The normal basic text editor properties are supported. **Select File | Open** to import a text file into the Info tab, overwriting the current content. Select **File | Save** to save the contents of the User Info area to a text file. To print the User Info, select **File | Print**. To get a preview of how the user info will look printed out, select **File | Print preview**. You can cut, paste, and insert text using the Edit menu. To change the look and style of the text, use the Format and Style menus. From here you can change fonts, font size, font color, and change the alignment of the text.
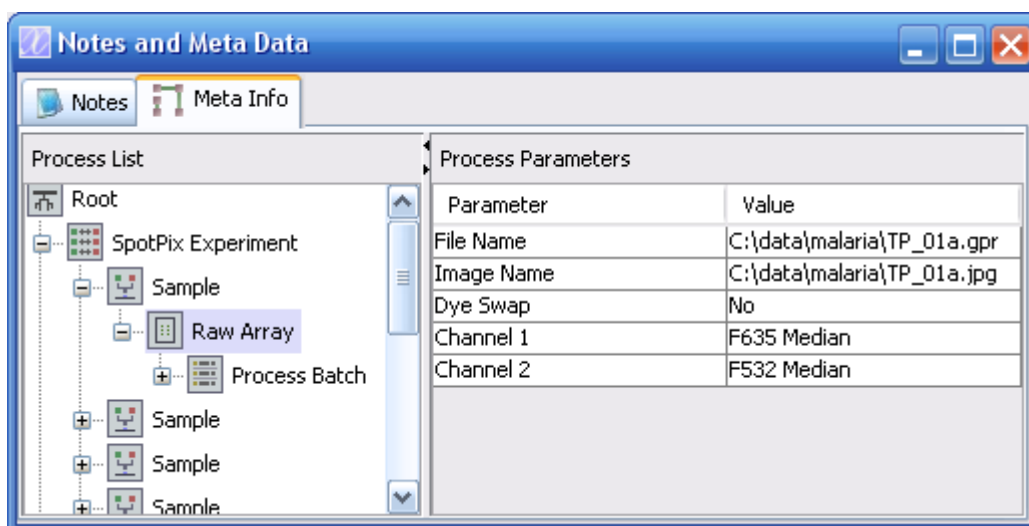
The **Meta Data tab** provides information on how a particular dataset was generated in J-Express Pro. The information includes source data file, how the data was imported into J-Express Pro, and how the subset was generated, if applicable. This feature helps documenting how the analysis results were obtained in J-Express Pro, and makes any saved analysis easy to recreate for others. Each dataset in a project has its own Process

List. As new datasets are added to the project tree, the Process List will be updated with the proper icon reflecting how the dataset was generated. The Process Parameters and their values of each process can be viewed by clicking on the icons in the Process List.

3.3.2 Showing / hiding project workspace windows

If you close any of the project workspace windows, selecting **Settings|Windows|Project Tree/Thumb View/Info|Show** from the J-Express menu bar will bring the respective window back. Choosing **Hide** from the same menu will hide the respective window, whereas choosing **Reset size and location** will reset the window to its default size and position. To reset all project workspace windows to their default size and position select **Settings | Reset All Windows**.

3.3.3 Changing colors and fonts

You can change the colors used for displaying profile values throughout J-Express Pro. To do this, click the **Fonts and Color Settings** button ( ) from the J-Express Pro tool bar, or select **Settings | Options** from the J-Express Pro menu bar. This opens the Settings window.
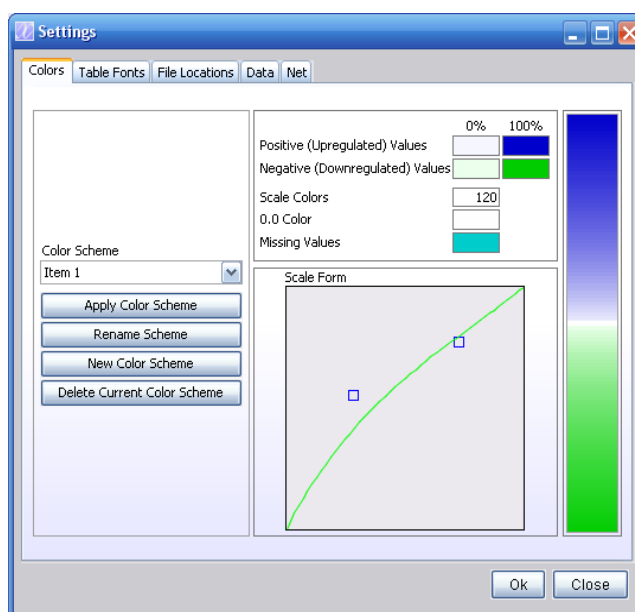
The Settings window contains four tabs: Colors, Table Fonts, File Locations and Data.

The **Colors tab** lets you select the colors used for displaying profile values. The four topmost color selection boxes are used to select the colors used for positive (upregulated) and negative (downregulated) values respectively. The **0%** boxes sets the colors to be used when a value is close to zero, and the **100 %** boxes set the colors to be used when a value is close to the maximum/minimum values of the dataset.

Scale Colors - the number in this box sets the number of colors for each color scale. This affects how coarse/fine each color scale is. A larger number creates a more smooth color scale; a smaller number produces a more coarse color scale.

0.0 Color - this color selection box allows you to set the color used to display zero values. Click the box to change to color**.**

Missing Values (Dendrogram only) - this color selection box allows you to change the color used for displaying missing (replaced) values in a dendrogram.

Scale Form - The color curve defines how quickly the color scale changes from the minimum value color to the maximum value color. Move the two blue boxes to alter the color curve. To have a completely linear color curve move the boxes to the center of the color curve area.

Changes made to the color curve are shown on the right side of the window, allowing you to interactively alter the colors used to suit your needs.

The **Table Fonts tab** lets you change the fonts used on info tables in J-Express Pro. The **Sample** area shows the look of the font you currently have selected. To change the font used select new fonts from the **Font** pull down menu. Select the style of the font (i.e. plain, **bold**, *italic*, ***bold italic***) from the **Style** pull down menu. Change the size of the font used by changing the value of the **Size** box. Check the **Use Group Colors** box to show info column text in the color of the group a profile is a member of.

You can create multiple color schemes and save/load them between projects.

The **File Locations tab** allows you to set the paths to Plugins, Pathways, J-Express Root, Libraries, SpotPix Files, Chromosomes Files and User Setting File. If several users share certain files, the paths to common repositories can be set here.

- Pathways - set the path to files used in J-Express Pro Pathway Analysis.

- J-Express Root - if J-Express is installed at one place for several users, you can set the J-Express Root path.

- Libraries - set the path to libraries here.

- SpotPix Files - set the path to SpotPix files.

- Chromosome Files - set the path to files containing chromosomal coordinates, used in Chromosome View.

- User Setting File - set the path to user settng file.

- GO Files – Gene Ontology analysis files
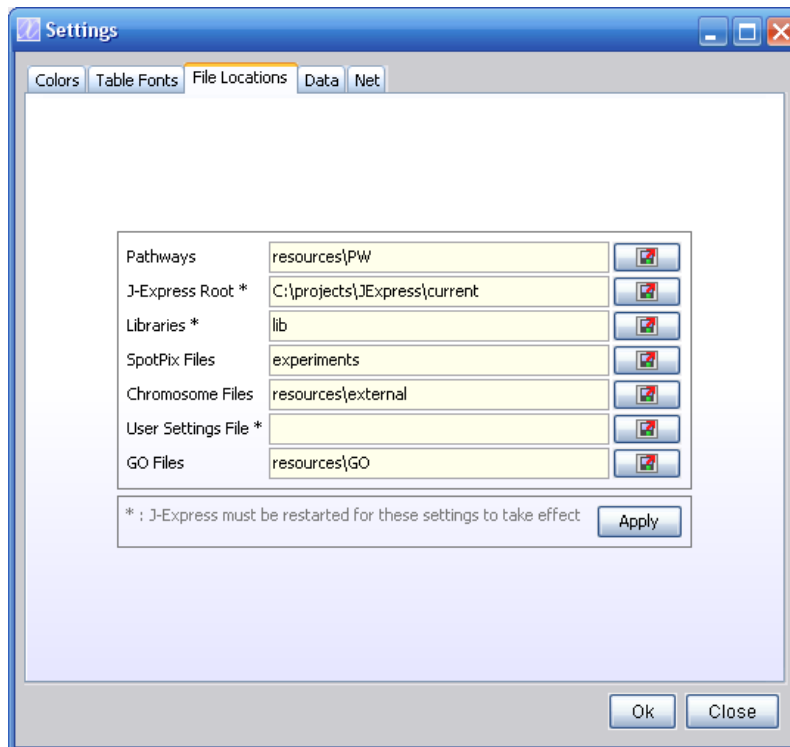
The **Data tab**
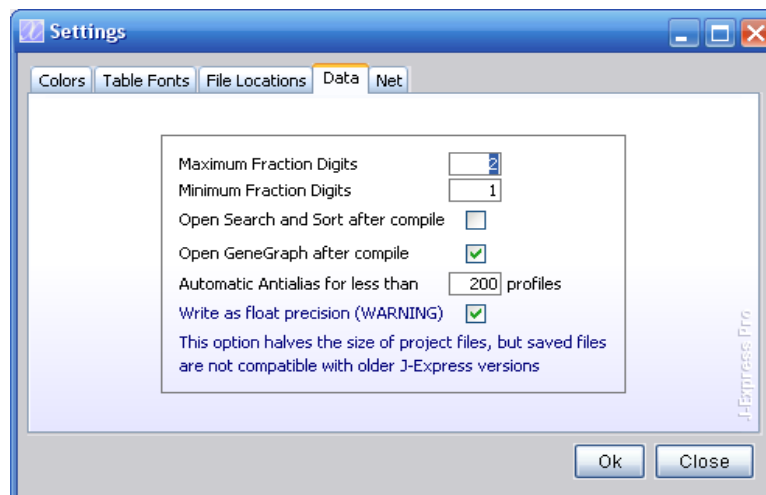
- Maximum Fraction Digits - The maximum number of fraction digits to use in all J-Express charts

- Minimum Fraction Digits - The minimum number of fraction digits to use in all J-Express charts

- Open search and sort, and open genegraph after compile – opens the corresponding window after a compile in the spotpix suite

- Automatic antialias – when opening a genegraph, antialisasing takes a long time to perform for many genes. Setting this value will automatically apply antialisasing if the number of profiles to display is below this number.

- Write as float – store the data values as 32 bit float numbers instead of 64 bit doubles. Some precision may be lost from the numbers, but the size on disk will be half for the data values. Generally, 32 bit float has more than enough precision for microarray data.

Click the **OK** button to use the current settings, and click **Close** to close the Settings window.

3.3.4 Saving Projects and Exporting data

To save an entire Project:

1. Click the **File** button ( ) on the tool bar, or click either **File** or **Project** from the J-Express Pro Menu bar.

2. Select **Save Project**. In the dialog that appears, browse to the directory where you wish to keep the project, enter a file name, and click **OK**. The project tree and all Info/metadata for nodes in the Project tree will be saved.

Save tabular data:

1. Select **Save Tabular**. In the dialog that appears, browse to the directory you wish to keep the exported data, enter a file name, and click **OK**. This will export the selected dataset into a tab-delimited format text file, with all available information and identifier areas, as defined in the Load External dialog. Information and meta data from J-Express Pro is <u>not</u> exported.

3.3.5 Creating and Managing Groups

The gene groups in J-Express Pro allow you to highlight sets of profiles that are of interest. Group membership is indicated in the program by its color. Groups can be created from selections made in all the tools in J-Express Pro. Group management is handled through two windows: Create Groups and the Group Controller.

You can create groups of genes or groups of samples.
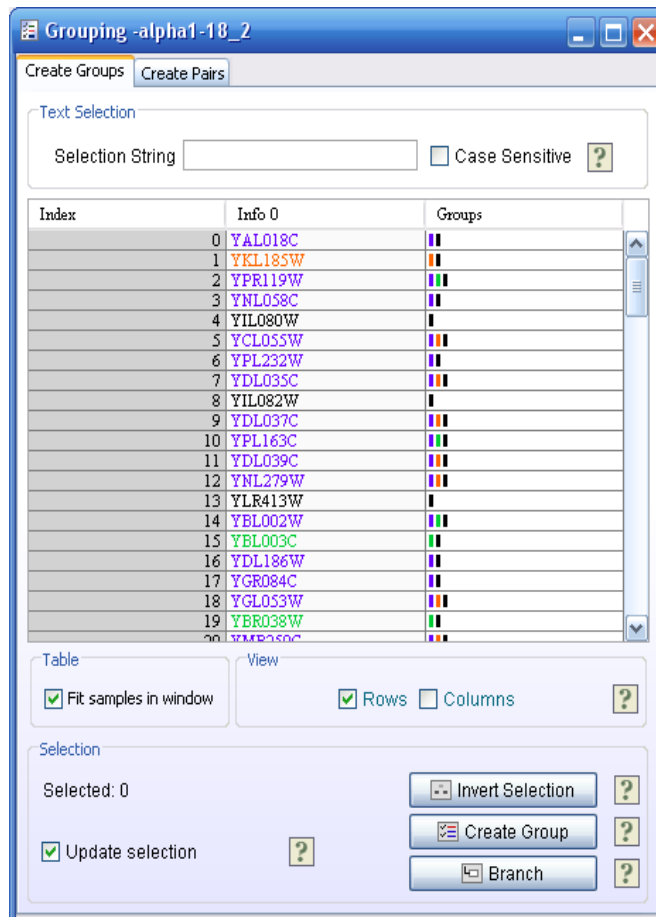
The Create Groups window provides a direct way of creating a new group.

Creating a group from scratch:

Bring up the Create Groups window by clicking the [icon] button (**Create Groups**) on the J-Express Pro toolbar, or selecting **Methods | Create Groups** from the J-Express Pro menu bar. You have several ways to select the profiles you want to include in the group.

To select profiles based on a shared prefix in the information columns (or select a single profile by its name), enter the name or prefix in the **Selection String** text field. For instance, to select all profiles starting with YLW enter `ylw` in the Selection String field. To differentiate between uppercase and lowercase names, check the **Case Sensitive** box (more advanced grouping trough a text search can be done trough the search and sort component). You can switch between creating row groups and column groups by checking and un-checking the **Rows** and **Columns** boxes.

Alternatively you can select profiles directly from the list by clicking on them. To select several consecutive profiles simply click and drag in the list or select the first profile you want to select, scroll to the last profile you want to include and then hold down shift on the keyboard while clicking it. To remove profiles from a selection, select them using the methods described. To finish creating groups select a color for the group's highlight by clicking on the **Color** button and selecting a

color from the color selection dialog that appears. Then click the **Create Group** button.

Another group can be created containing all the profiles not in the first group. Click the **Invert Selection** to select all profiles not selected and unselect all profiles that were first selected. Give this group a different color and click **Create Group.**

A group can also be branched off to give it its own node in the Project Tree. To do this simply click the ⬒ button. A new node will be added below the current one in the Project Tree labelled with the symbol ⬓.

The group can be given a certain color, name and description trough this interface. To show in the different charts, the group must be active. All these properties can be changed trough the group controller later.
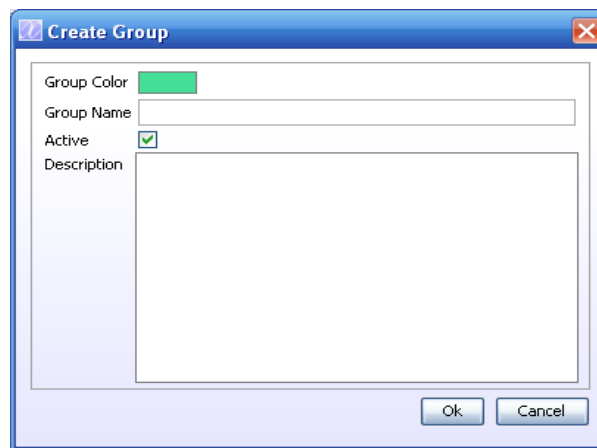
Creating Groups from an analysis window:

Using the group functionality in a dendrogram or selecting a cluster in the K-Means clustering window both define a subset of profiles that can be used to create a new group. In general, all functions that result in the creation of a new tab in a function window (such as zooming on a branch of a Hierarchical Clustering tree) can be used for creating new groups.

3.3.6 Defining sample pairs

Paired analysis require you to first define which samples belong together in pairs.

1. Select **Create Groups / Pairs** from the **Data Set** menu or click on the ⬛ (**Create groups / pairs**) button.

2. Click on the **Create Pairs** tab.

After generating the tab you wish to create a group for, make sure the tab containing the data is selected, and then click the *Create Group* (⊞) button on the toolbar of the window containing the tab. To manage all groups created, open the *Group Controller*, by clicking the ⊞ button on the J-Express Pro main window toolbar.
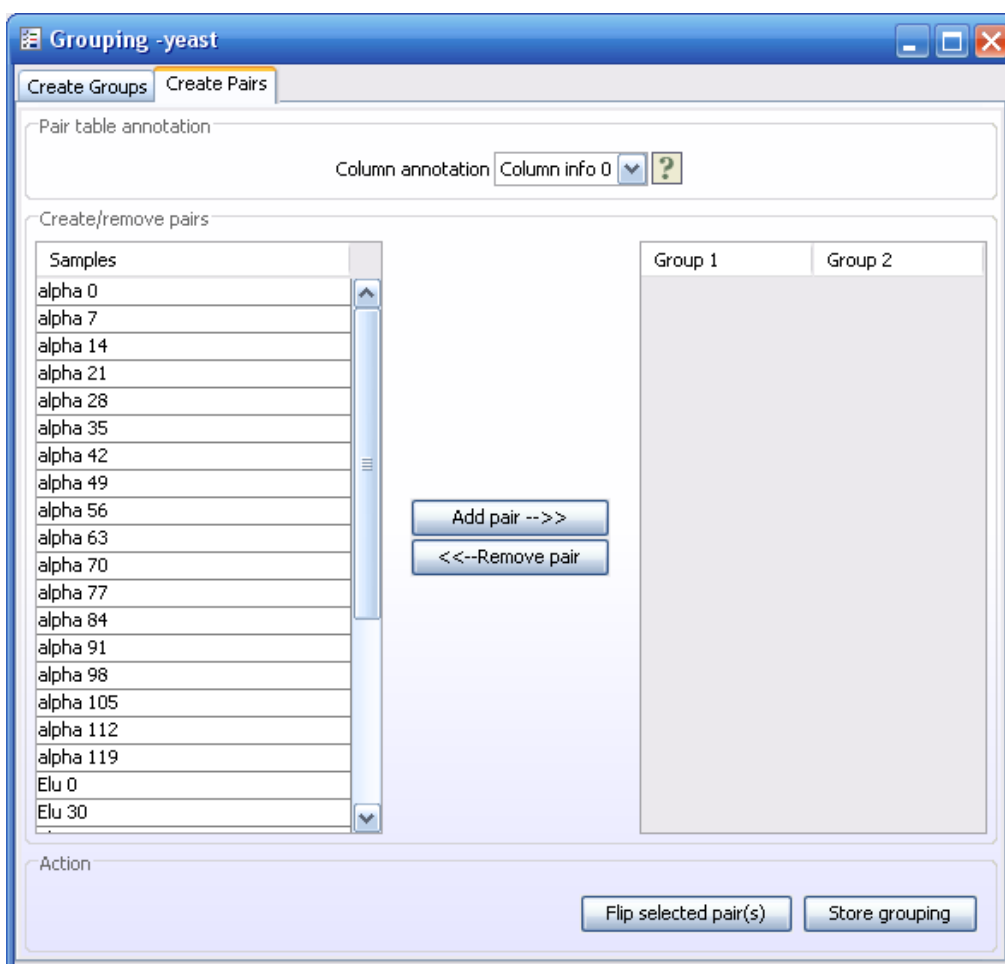
1. Press the Ctrl button and click on the two samples you want to pair up. If you are using mac, you should go to the Data Set menu and rearrange the columns so that the pairs are next to each other in the list before creating pairs.

2. Click the Add pair button. The paired samples will now be removed from the Samples list and added to the paired list.

The direction of the pair should be the same for all pairs. If some of the pairs appear in the wrong order, select them and click **Flip selected pair(s)**

3.3.7 Managing Groups:

The Groups window contains a list of all created groups. In addition the number of profiles contained in each group is shown in the column labelled "Count". The list is hierarchical, so if a profile is a member of several groups, the topmost group membership is the one that is applied to the profile when displayed. For instance, if a profile is a member of both groups 1 and 3 in the image above, then J-Express Pro will display the group as a member of group 3 since this group is higher in the list.
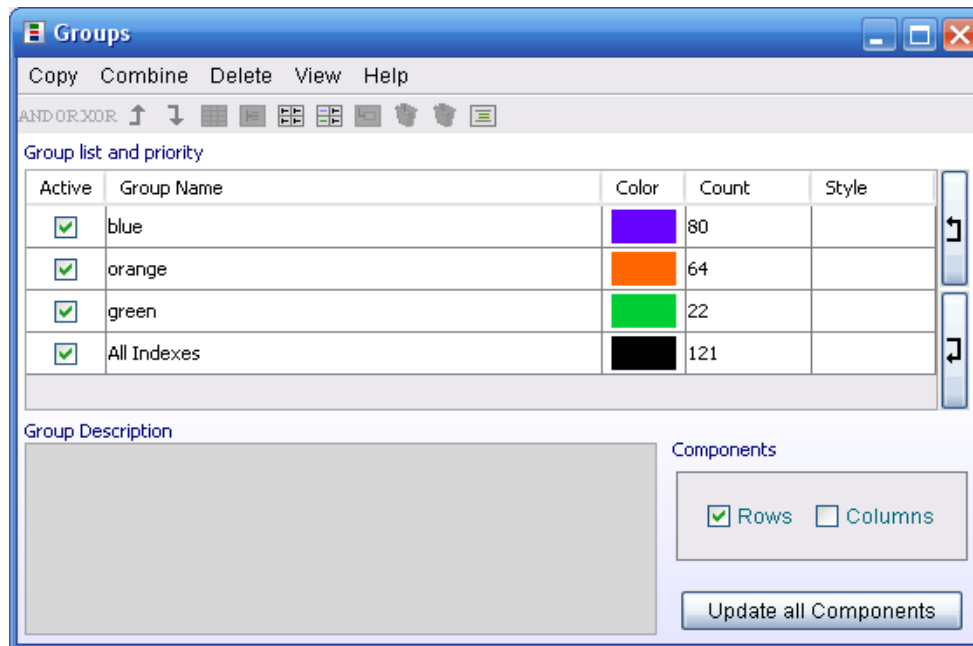
Change the name of a group by double-clicking the **Group** Name entry and entering a new one. To move a group up or down in the list use the ⤴ and ⤵ buttons to the right of the list.

If you want to temporarily disable the highlighting of a group, uncheck the **Active** box to the left of the group's name in the list. To re-enable the highlight for the group, check the Active box again.

To change the color of a group's highlight, click the **Color** box of the group. A color selector dialog opens that lets you choose a new color .

To copy the groups to children nodes in the Project Tree select the **Copy Group to Children** (⤓) button.

To copy the groups to the parent node in the Project Tree click the **Copy Group to Parent** (⤒) button.

To perform a logical AND operation on groups, select the groups, and then click the **AND** button. This will create a new group containing only the profiles that are members of *all* the selected groups.

To perform a logical OR operation on groups, select the groups, and then click the **OR** button. This will create a new group containing the members of all the selected groups.

To display the contents of a selected group in a spreadsheet, click the **Show group in table** button (🔲). This will open a spreadsheet window containing the data of all the profiles in the selected group.

To display the contents of a selected group in a Gene Graph window, click the **Show group in graph** button (🔲). This will open a gene graph window containing all the profiles of the selected group.

To display thumbnails of all defined groups, click the **Show all groups as thumbs** button (🔲). This will bring a window similar in function to the K-means window showing thumbnails of all defined groups. See section 3.4.2 for additional informati on.

To branch the data contained in a selected group into a sub-node of the project tree, click the **Branch data** button (🔲).

To remove groups from the list select the group you want to remove by clicking on its entry in the list, and then clicking the **Delete Group** button (🗑).

To remove groups and the profiles associated with the group from the data set completely, select the group from the list, and click the 🗑 **(Delete Group From Data)** button.

The information contained in the group controller can be written to a **Group Legend** by clicking the 🔲 button, and then saved.

All these functions are also available from the Group Controller menu bar.

Note: Changing group color and hierarchy does not take effect until you click the **Update All Components** button.

Click the **Close** button to remove the Groups window.

Note on selections:

If you have several windows open, profiles you have selected in one window will remain selected in the others. For instance, if you can select a profile of interest in a Hierarchical Clustering Window, the profile will be automatically selected when you open a Find Similar Profiles window. If changes you make do not take effect immediately, press the 🔲 button (**Update and Repaint**). Additionally the selected data can be shown in a Gene Graph viewer simply by opening a new Gene Graph viewer with data selected.

3.4 The Gene Graph Viewer

The Gene Graph Viewer provides a detailed graphical and interactive view on a set of expression profiles. Several profiles can be shown at one time in a Gene Graph window allowing visual comparisons to be made between profiles. The graphs can be exported as images or as HTML files, and additional information on a particular profile can be obtained by searching external Internet databases from within J-Express Pro. The Gene Graph window is often used to provide additional information obtained from the other analysis methods available to you in J-Express Pro.

Note: Units on a Gene Graph are scaled and optimized for showing all profiles in the selected dataset and the parents (ancestors) of the data set in the project tree. To optimize the units scale for a particular subset of data use the **Clone Dataset to Root** function, or uncheck the **Scale relative to parents** box in the project tree window.

3.4.1 Opening the Gene Graph Viewer

1. Select the dataset you want to display in the Project Tree by clicking its node.

2. Click the ▦ button on the J-Express Pro tool bar or select **Methods|Gene Graph Viewer** from the J-Express Pro menu bar.

A Gene Graph window will open displaying the selected dataset as graphs.
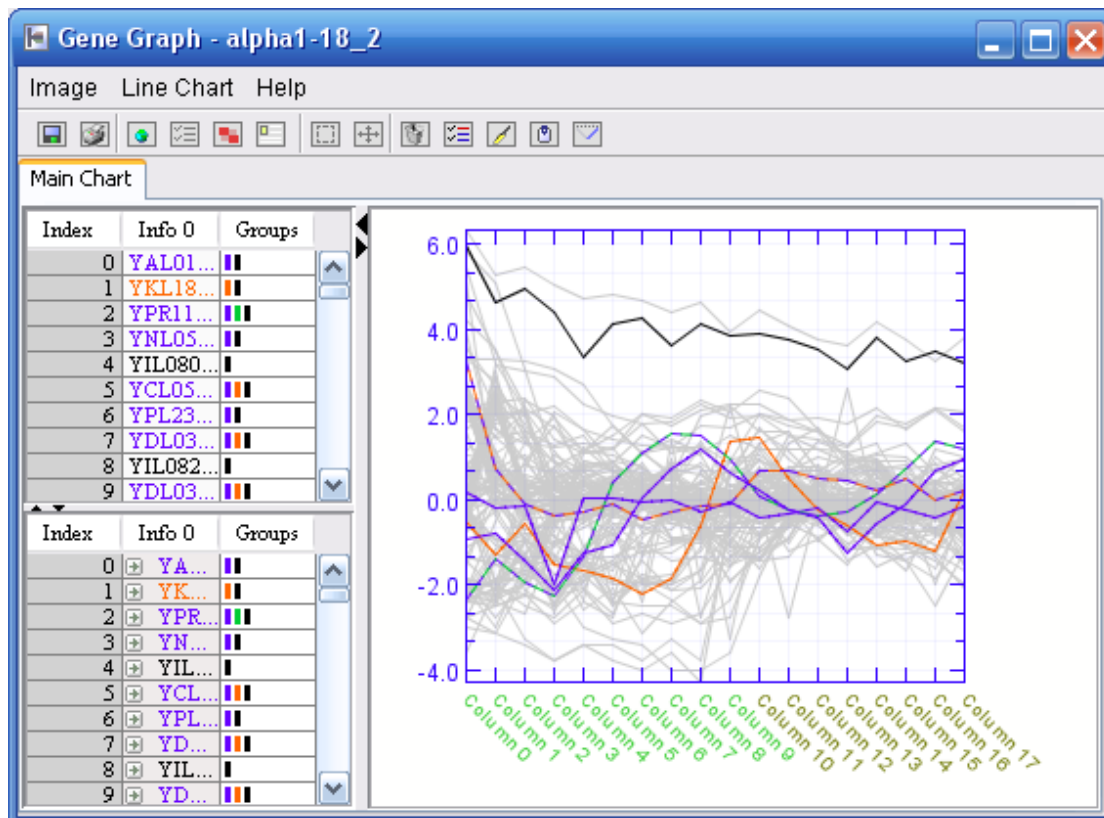
A Gene Graph window showing multiple profiles, with Shadow Unselected turned on, and External Links window open.

3.4.2 Modifying the Gene Graph display

The gene graph window provides many ways to control how the profiles are displayed. The following section will familiarize you with all visual functions of the Gene Graph.

When showing multiple profiles in the same Gene Graph window it may be difficult to separate one profile from another. Using the Shadow Unselected feature of J-Express Pro you can highlight profiles of your choice to bring out interesting features of a graph.

How to use the Shadow Unselected feature:

1. From the list of profiles in the left part of the Gene Graph window select the profile you wish to highlight. To select multiple profiles lying next to each other in the list, select the first profile you want to highlight, scroll to the last profile you want to highlight, hold down the shift key and click the last profile. All profiles lying between the two will be selected. To select multiple profiles that lie separated in the list, hold down the control key on the keyboard while clicking on each profile to select them. You may also select genes in another window, *e.g.* a dendrogram window.

2. Click the ▦ button (Shadow Unselected) or select **Line Chart|Toggle Shadows** from the Gene Graph window menu bar. The selected profiles will be shown in full color, while the other profiles will fade to a gray color.

Using the External Links List:

Click the 🌐 button (**External Links List**) on the Gene Graph tool bar. This splits the selection window of the Gene Graph in two, with matching content. The top window is used for selection as normal; the bottom window contains hyper-links to the default external database on the World Wide Web. Click one of the rows in this window to open a web browser and perform the external search.

Improving Graphical Quality:

Click the ▦ button (**Antialiasing**) on the Gene Graph tool bar, or select **Line Chart|Toggle Antialias** from the Gene Graph menu bar. The aliased (jagged) edges on the graphs and text will disappear.

Note: on large datasets this function can be time-consuming. If you experience long pauses while refreshing or generating displays we suggest turning antialiasing off.
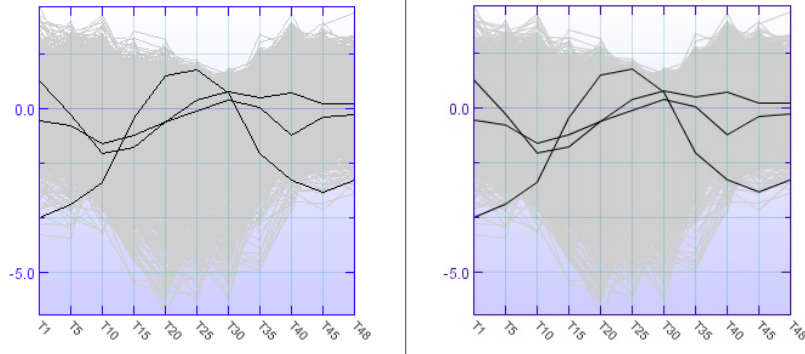
Antialiasing. The image on the left shows a graph in normal mode, while the image on the right shows the same graph in antialiased mode.

Put in Tree:

To place the entire component into the project tree click the ⬛ button, **Line Chart|Put in Tree** from the menu bar. This creates a new node in the project tree that acts as a direct shortcut to the current component.

Creating a HTML version of a graph:

To create a HTML version of a graph for display on a web page click the ⬛ button (**Export to HTML**) on the Gene Graph toolbar. In the file location dialog that appears, locate the folder you want to save the HTML version of the graph, enter a filename, and click **OK**. Make sure you give the file the suffix `.htm` or `.html` or you will be unable to open the file in your web browser.

The HTML page generated contains the time/date the page was generated, an image of the graph, and a list of all profiles that that are shown in the graph. If Shadow Unselected is active then the selected profiles will be shown in bold typeface in the list.

Zooming a graph:

To zoom in on an area of interest in a graph, click the **Zoom in** button (⬛), and drag out the area you want to zoom in on. A new tab will be created in the Gene graph window containing the zoomed view. To dispose of the zoom tabs, click the **Remove component** button (⬛).

Move:

The move button (⬛) enables you to grab the graph window using the mouse cursor instead of using the scrollbars (if the graph is too big to fit in the window).

Create Group:

The **Create Group (** ⬛ **)** button will create a group containing the selected genes that can be managed further from the Group Controller (see section 3.1.14).

Repaint Component:

If changes you make do not take effect immediately, press the **Repaint Component (**) button or select **Line Chart |** Update and Repaint

Copy image to Clipboard:

To copy the image in any of the tabs to clipboard, click the button.

Put in Tree:

To place the entire component into the project tree select **Line Chart|** **Put in Tree** from the menu bar. This creates a new node in the project tree that acts as a direct shortcut to the current component.

Color Mode

Full color Mode - will display all component and group colors in the set color.

Black and White Mode - will display component and group colors in a shade of grey.

Saving a graph as an image:

To save a graph as a separate image file click the button (**Save chart**) on the Gene Graph tool bar, or select **Image|Save** from the Gene Graph menu bar. In the file location dialog that appears, locate the folder you want to save the image of the graph, enter a filename, select the appropriate file extension, and click **OK**.

Printing a graph:

To print graphs simply press the button (**Print**) on the Gene Graph toolbar, or select **Image|Print** from the Gene Graph menu bar. <u>Note</u>: it is recommended to print graphs using the Landscape paper orientation, since graphs usually are larger horizontally. Consult your printer and/or operating system manual for information on how to print using Landscape paper orientation.

Customizing the appearance of the Gene Graph:

Right clicking on the graph or selecting **Line Chart|Chart Layout** will bring up the Plot Properties dialog. Here you can alter most visual aspects of the Gene Graph.
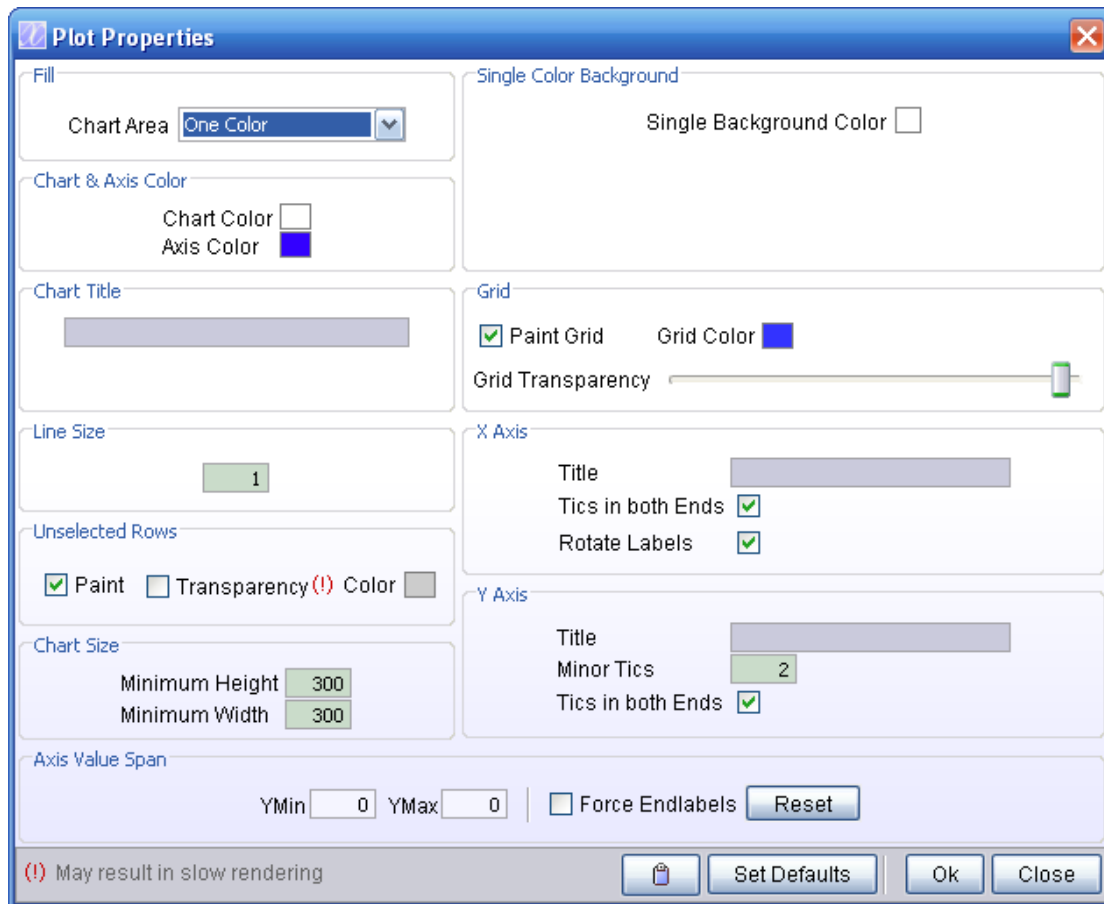
Plot Properties dialog.

The **Fill** menu allows you to choose the appearance of the background of the plot. The various options are:

- One Color – Single color is used for the background. Click the colored square to the right of the menu to choose the desired color.

- Gradient – Two colors are combined to create a smooth color gradient. Click the two colored boxes to choose the desired colors. Use the Gradient Type menu to select the type of gradient. Diagonal forms a color gradient from the upper left to the lower right corner; Top-Bottom forms a color gradient from the top of the plot to the bottom.

- External Picture – Use the file selection dialog to select the image file you wish to use as a background for the plot. Selecting Stretch will stretch the image to fit the plot. Selecting Tile will repeat the image in a tile pattern if it is to small to cover the entire plot.

- Tiles – Six additional patterns you can use for your plots.

The menu to the right of the Fill menu, is a menu that is linked to the options chosen in the Fill menu. If gradient is chosen the two colors can be selected in this menu. Likewise is External Picture is chosen, the picture path can be set in this menu.

Chart Title lets you create a title text for the plot that will be displayed at the very top of a plot.

Line Size sets the thickness of the line used for drawing the graphs, in pixels.

Chart Size lets you set the minimum height and width of the plot, in pixels.

Unselected Rows lets you set options for the use of the Shadow Unselected feature.

- **Paint** – uncheck this box to disable the display of the unselected profiles.

- **Transparency** – uncheck this box to use a solid color for the unselected profiles. If checked, the color used to display unselected profiles will be partially transparent, showing part of the background color through. Note that the use of transparency may result in lower performance on slower systems.

- **Color** – Click the color box to choose the color used for the unselected profiles.

Axis Value Span - select the minimum and maximum value of the Y-axis that you are interested in looking at.

- **Force Endlabels -** check to round the minimum and maximum values upwards (positive values) or downwards (negative values) to the closest value that can be divided by 5. The rounded minimum and maximum values will be forced to be at the bottom and top, respectively, of the diagram.
- **Reset button** resets the minimum and maximum values to default.

Grid lets you set options for the plot grid.

- **Paint Grid** – check this box to toggle display of the grid on. Uncheck it to toggle display of the grid off.

- **Grid Colors** – select the desired color for the grid by clicking on this box and choosing a color from the dialog that appears.

- **Grid Transparency** – Use this slider to set the transparency of the grid, relative to the background.

X Axis lets you set options for the appearance of the x-axis of the plot

- **Title** - lets you set a title for the x-axis, which will be shown at the bottom of the plot.

- **Tics in both Ends** – check this box to have unit tics at both the top and bottom of the plot. If left unchecked, unit tics will only be used along the bottom of the plot.

- **Rotate Labels** – Check this box to rotate the text for the state labels by 45 degrees.

Y Axis lets you set options for the appearance of the y-axis of the plot.

- **Title** – lets you set a title for the y-axis, which will be shown on the left side of the plot.

- **Minor tics** – enter the number of minor tics you want between every major tic along the y-axis in this box.

- **Tics in both ends** – check this box to have unit tics at both the left and right side of the plot. If left unchecked, unit tics will only be used on the left side of the plot.

3.5 Hierarchical Clustering
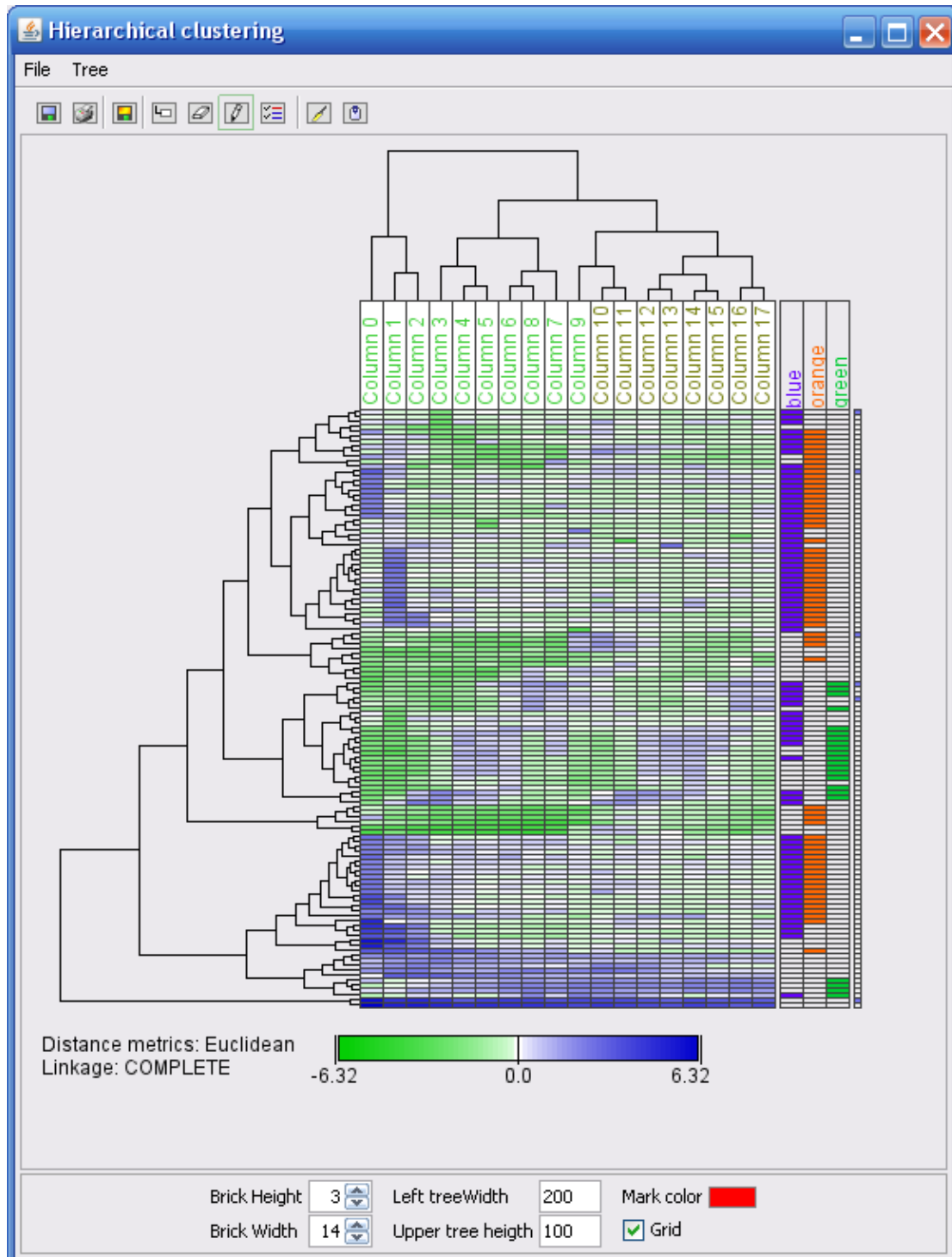
A dendrogram.

3.5.1 The Hierarchical Clustering Window

Select the node you want to analyze in the Project Tree and click the button (**Hierarchical Clustering**) on the J-Express Pro tool bar. Alternatively, select **Methods| Hierarchical Clustering** from the J-Express Pro menu bar.

The dendrogram consist of several components. On the left side is the generated clustering tree for the set of data shown in the tab. A similar clustering tree is generated for the **columns** of the data if the **Cluster Columns** option is selected. This tree will then be shown above the main dendrogram. The dendrogram itself is arranged according to the result of the hierarchical clustering. Each row of squares represents one profile in the dataset. The color rectangles makes up the heat map and gets the colors from the global color scheme. Select different colors at Settings -> Options -> Colors.

The identifiers of each state are placed along the top of the dendrogram, if the **ID Row** was defined during data loading.

Group membership is indicated with colored boxes immediately to the right of the dendrogram profiles. Note that all group memberships for a profile is shown. Group names are shown at the very top of each column of colored squares for a particular group. To the right of the group columns the External Information for each profile is shown, as defined during the data loading process.

Branch Properties (colored square):

Pointing the mouse cursor over the root of a subtree will highlight this subtree with the color specified by the mark color (default color is red). To change the color click on the colored square and choose a new color.

The different mouse-click modes

> **Mark subtree/Set Branch color (** **):**
>
> To mark a subtree, press the **Set Branch color (** **)** button to set the color specified by the *Mark color*. Then point the mouse cursor over the root of a subtree, and click on it. This will mark the subtree with the specified color.
>
> **Remove Branch color (** **):**
>
> If the **Remove Branch color (** **)** button is clicked it is only possible to highlight a sub-tree with the specified color. Clicking on a sub-tree will not mark it.
>
> **Branch (** **)**
>
> Click this button to select the branch mode. When in branch mode, you can click a branch to create a new dataset containing the genes below this branch.
>
> **Create group(** **)**

Use this mode to create a group of all the genes below a branch.

Copy image to Clipboard (🖼):

To copy the image in any of the tabs to clipboard, click the **Copy image to Clipboard (🖼)** button.

Store result in project

To place the entire component into the project tree select *store result in project.* This creates a new node in the project tree that acts as a direct shortcut to the current component.

3.5.2 Setting options for Hierarchical Clustering
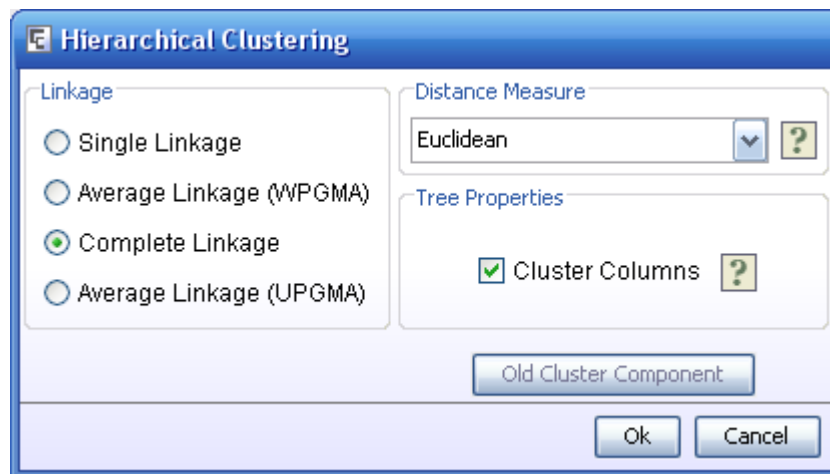
How dendrograms are generated is controlled from the Hierarchical Clustering properties.

It is recommended that you try alternative settings for Hierarchical clustering. Click the **OK** button to activate the new settings. Close the window by clicking the **Close** button.

Cluster columns:

To generate a clustering tree for the states/columns of the data, check the **Cluster columns** box. A Hierarchical Clustering tree will be generated for the data based on the columns. The tree will be shown to the top of the dendrogram.

Weighted twigs:

J-Express Pro uses weighted twigs to generate the Hierarchical Clustering tree by default. This means that the horizontal length of a twig is then based on the distance between the sub-trees joined by these twigs (*e.g.*, the distance between two expression profiles if the twigs connect two single profiles).

Linkage:

Select the desired Linkage Method from the list given. For explanation of the effect of the different linkage methods, see Section 4.2.2

Distance Measure:

To choose a different distance measuring method, choose a new one from the **Distance Measure** list. For definitions of the different distance measures, please refer to Section 5.1.

Visual Dendrogram properties:

At the bottom of the dendrogram window, you can set the layout of the dendrogram.

Tree Height and Width:

The tree on depicted to the left represents row wise clustering of the data. The upper tree represents column wise clustering of the data.

Upper Tree Height:

The number in this text field indicates the height of the columns clustering tree in pixels.

Left Tree Width:

The number in this text field indicates the width of the Hierarchical Clustering tree (row clustering) in pixels.

Brick Height

The height of each row. This will also impact the annotation as the annotation wil fit inside each row. If the brick height is too small (default) there will be no annotation in the window.

Brick Width

The width of each color brick in the heat map.

Mark color

When in branch color mode (by clicking the pencil at the top menu) you can select the color to give clicked sub-trees by clicking this color button.

Grid

Toggle on and off a grid.

3.6 Hierarchical clustering with distance matrix
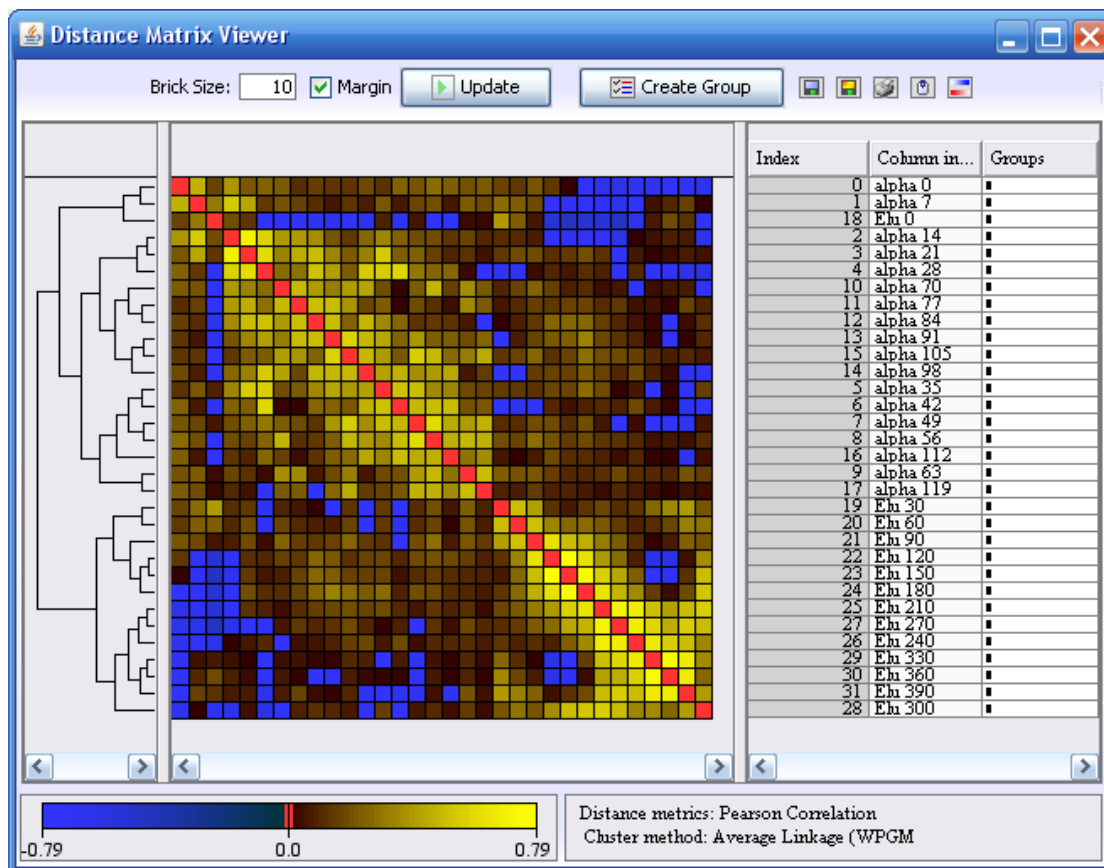
The Distance Matrix View component can be used to discover genes that have correlated expression patterns.

3.6.1 The Distance Matrix Viewer Window

Select the node you want to analyze in the Project Tree and click the ⬚ button (**Hierarchical Clustering**) on the J-Express Pro tool bar. Alternatively, select **Methods | Hierarchical Clustering With Distance Matrix** from the J-Express Pro menu bar.

The **distance matrix viewer** displays a distance matrix correlation map in the center of the window. The distance matrix shows the distance between the expression profiles of all genes in the dataset. The color of each square reflects the distance between the corresponding profiles. The color map and its maximum and minimum values is shown in the lower right hand corner of the window. The red diagonal line shows the distance of a profile to itself. The matrix is symmetrical about the red diagonal.

The **hierarchical clustering tree** is displayed to the left of the window. The genes in the matrix viewer are ordered by the hierarchical clustering tree. Hence, profiles with small distances, ie high correlation, will be adjacent in the matrix viewer. Several adjacent genes with highly correlated profiles will appear as larger, yellowish squares at the diagonal.

The spreadsheet to the right of the distance viewer contains **additional information** about the genes, if available.

Use with the Gene Viewer:

Open the gene viewer by clicking on the **Line Chart (**   **)** button on the J-Express Pro main toolbar, or select **Methods | Gene Graph Viewer.** Press the **Shadow Unselected (**   **)** button on the Gene Graph Viewer menu bar. Next select genes in the Distance Matrix Viewer by clicking and dragging the mouse in the distance matrix. The expression profiles of the selected genes can now be seen in the Gene Graph viewer. To select more than one area, press Ctrl + click and drag.

Brick Size:

The **brick size value** determines the size of the coloured squares in the distance matrix viewer. The clustering tree at the left side will resize to fit the brick sizes.

Margin:

Check this box to draw a line around the squares in the distance matrix. Note: if the brick size value is 1, the margin box should be **unchecked**. Otherwise the entire matrix will be black.

Update:

If changes you make do not take effect immediately, press the update button.

Create Group (**):**

To create a group containing certain genes, click and drag the mouse in the distance matrix, and then press the   **Create Group** button. To create a group containing genes that are not immediately next to each other, press Ctrl + click and drag mouse, for each area you want to include. Then press the   **Create Group** button.

Save Chart (**):**

There are three components that can be saved as an image. These are the distance matrix, color scale and the spreadsheet. To save any or all of these, click the   button (**Save Image**), and select the components you want to save. All selected components will be framed to the same image. Click **OK**. In the dialog that appears, locate the folder you want to save the image in, enter a filename, and choose an file extension from the pull down menu. Note that in addition to regular image file options, it is also possible to save the image as Scalable Vector Graphics(.svg). This is very useful if you want to zoom in on certain areas of the image and still retain the same picture quality.

Print Chart (**):**

There are three components that can be printed. These are the distance matrix, color scale and the spreadsheet. To print any or all of these, click the   button (**Print**

Image), and select the components you want to save. All selected components will be framed to the same image. Click **OK**.

Copy Image To Clipboard (🖼 **):**

There are three components that can be copied to clipboard. These are the distance matrix, color scale and the spreadsheet. To copy any or all of these to clipboard, click the 🖼 button (**Copy Image To Clipboard**), and select the components you want to copy. All selected components will be framed to the same image. Click **OK**.

Change Color Scale (🖼 **):**

To change the colors and color curve, press the **Change Color Scale(** 🖼 **).**

The four topmost color selection boxes are used to select the colors used for positive (correlation) and negative (anti-correlation) values respectively. The **0%** boxes sets the colors to be used when a value is close to zero, and the **100 %** boxes set the colors to be used when a value is close to the maximum/minimum values of the dataset.

- **0.0 Color** - this color selection box allows you to set the color used to display zero values. Click the box to change to color**.**
- **Scale Form** - The color curve defines how quickly the color scale changes from the minimum value color to the maximum value color. Move the two blue boxes to alter the color curve. To have a completely linear color curve move the boxes to the center of the color curve area.

Changes made to the color curves are shown on the right side of the window, allowing you to interactively alter the colors used to suit your needs.

3.6.2 Setting options for Hierarchical Clustering With Distance Matrix
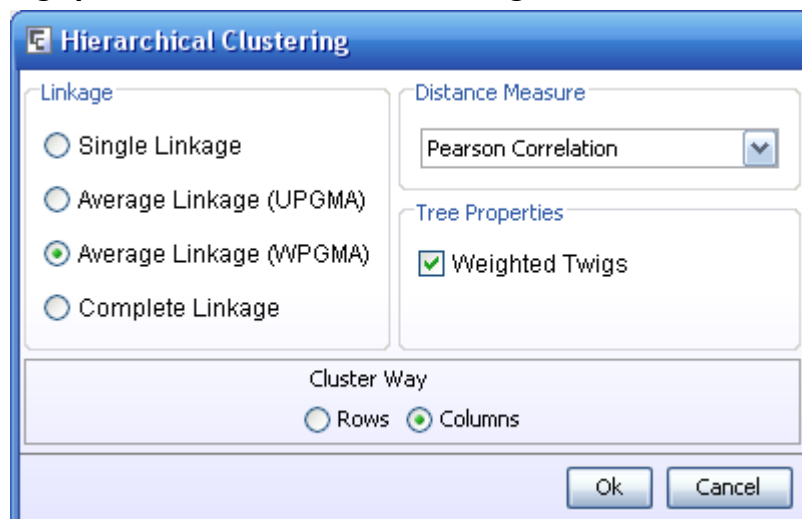
When pressing the **Hierarchical Clustering With Distance Matrix (** 🖼 **)** button or **Methods|Hierarchical Clustering With Distance Matrix,** the dialog window

printed above will appear. This window almost the same as the one for Hierarchical Clustering( ).

Linkage:

Select the desired Linkage Method from the list given. The linkage method chosen specifies how distances are calculated between clusers.

Distance Measure:

To choose a different distance measuring method, choose a new one from the Distance Measure list.

Weighted twigs:

J-Express Pro uses weighted twigs to generate the Hierarchical Clustering tree by default. This means that the horizontal length of a twig is then based on the distance between the sub-trees joined by these twigs (*e.g.*, the distance between two expression profiles if the twigs connect two single profiles). Unweighted twigs use a constant horizontal length for the twigs, usually resulting in a wider tree. To use unweighted twigs, uncheck the weighted twigs box.

Cluster Way:

Pressing the Rows radiobutton will result in clustering of rows. Likewise, pressing the Columns radiobutton will result in clustering of columns.

It is recommended that you try alternative settings for Hierarchical clustering. Click the **OK** button to activate the new settings. Close the window by clicking the **Close** button.

3.7 K-Means Clustering
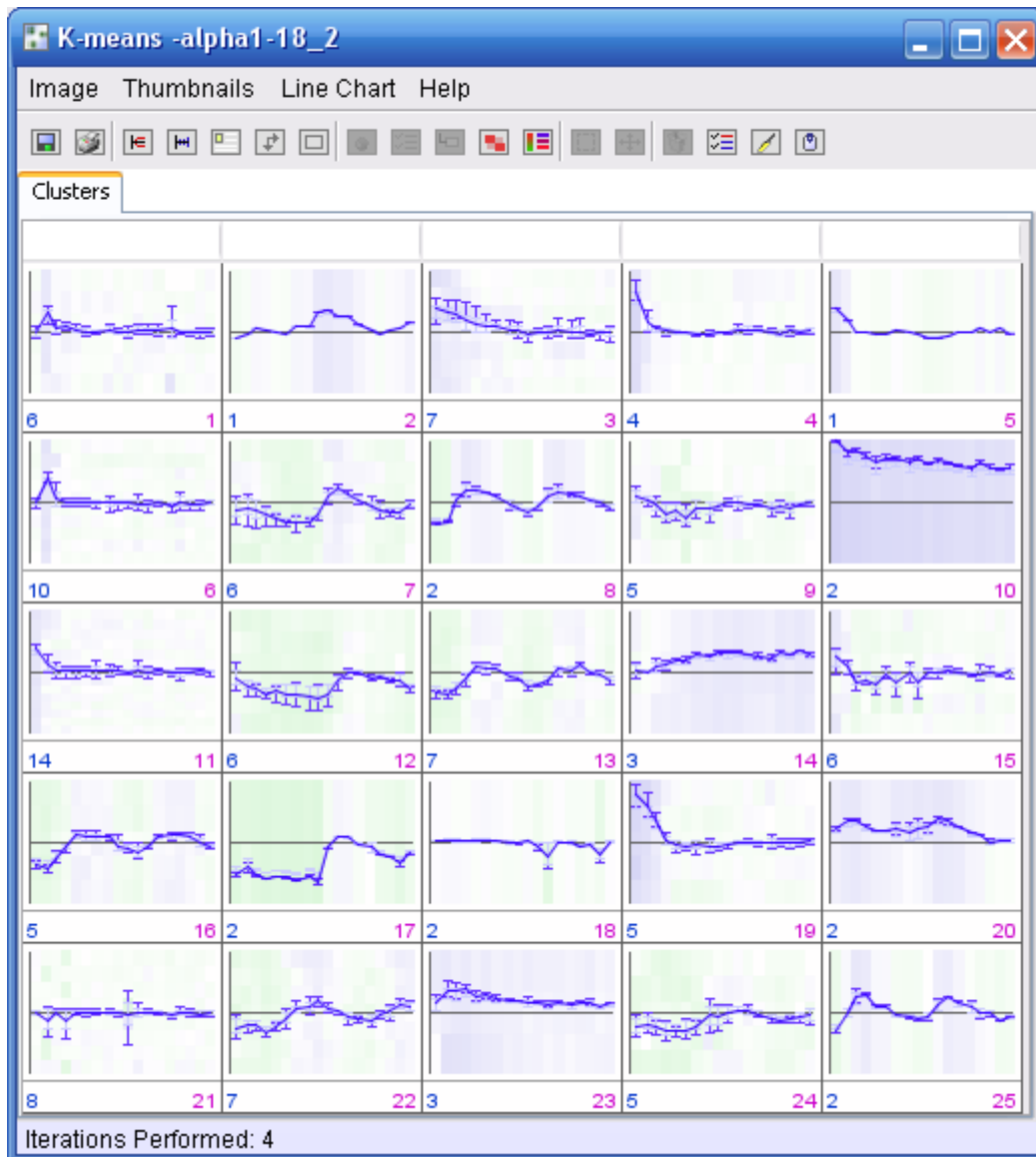
3.7.1 The K-Means Clustering Window

Select the node you want to analyze by in the Project Tree and click the ⬚ button (**K-Means Clustering**) on the J-Express Pro tool bar. Alternatively select **Methods | K-Means Clustering** from the J-Express Pro menu bar.
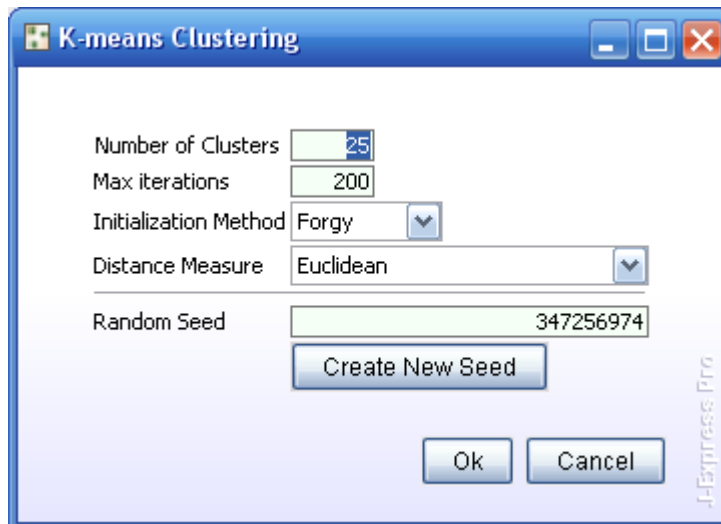
K-Means default properties.

The K-Means properties dialog appears. This dialog allows you to configure how the K-Means algorithm will operate.

Number of Clusters defines how the number of clusters/groups desired, *i.e.*, the number of groups the set of profiles should be split into

Max Iterations defines the maximum number of iterations to be performed in the K-means clustering. The algorithm may fail to converge, so a maximum number of iterations must be set. Clustering a large dataset (> 5000 rows) usually needs more iterations than a small one.

Random Seed is used as a basis for randomizing the algorithm. If you need to recreate a particular analysis exactly, entering the same random seed and keeping all other options the same will yield the same result. A random seed number can be any (large) number. Clicking **Create Random Seed** will create a number for you. The seed used is saved (together with all parameter values for K-Means) in the meta data for any data set resulting from the analysis.

Initialization Method allows you to choose from different initialization methods. To select one, pick one from the drop-down menu.

Distance Measure - allow you to choose different distance measuring methods, by picking one from the **Distance Measure** list. For definitions of the different distance measures see Section 5.1.

To set the current settings as default for K-Means Clustering in the future, click the **Set Defaults** button.

To run K-Means Clustering with the current options click **OK**, and the K-Means Clustering Window will open. This window follows the usual J-Express Pro pattern, with a menu bar, a tool bar and an area for data display organized by tabs.

The result of the K-Means Clustering is shown in the Clustering tab. This tab shows a number of thumbnails of graphs, one for each cluster. By default, each thumbnail shows the mean of the profiles contained in that cluster, and are marked with the id of that cluster. The number of profiles contained in that cluster is also displayed underneath each thumbnail. Clicking on the thumbnail will add a tab to the K-means window displaying a gene graph window. For more information on gene graph see Section 3.2

3.7.2   K-Means Clustering window Features

Show all profiles ( ):

To show all the profiles contained in the clusters, click the button (**Show all profiles**) on the K-Means window tool bar, or select **Thumbnails|Show All Profiles** from the K-Means window menu bar. To go back to showing the mean profiles click the button (**Show Mean Only**) on the K-Means window tool bar, or select **Thumbnails|Show Mean Profile Only** from the K-Means window menu bar.

Saving an image of the thumbnails ( ):

To save an image of the thumbnails, click the button on the K-Means window tool bar or select select **Image|Save** from the K-means menu bar. Locate the folder where you want to save the image by clicking the **Browse** button, set the appropriate file extension, and enter a filename. Click **OK.**

Printing the thumbnails ( ):

To print the thumbnails click the button on the K-Means window tool bar.

Export to HTML ( ):

To generate a HTML version of the thumbnails, click the button (**Export To HTML**) on the K-Means window tool bar, or select **Thumbnails | Export to HTML** on the K-Means window menu bar. Select a location and a name for the html file (remember to include the .html extension) in the dialog that appears. A subdirectory containing images for the web page will be created along with a HTML file that shows thumbnails of the clusters, and lists the contents of each cluster.

Antialiasing ( ):

To improve the visual quality of the thumbnails click the button (**Antialiasing**) on the K-Means window tool bar, or select **Line Chart | Toggle Antialias** from the K-Means window menu bar. The aliased (jagged) edges on the graphs and text will disappear.

Note: on large datasets this function can be time-consuming. If you experience long pauses while refreshing or generating displays we suggest turning Antialiasing off.

Toggle Colors ( ):

To use group colors in the graphs, click the button (**Toggle Colors**) on the K-Means window toolbar or select **Thumbnails | Toggle Line Color** from the K-Means window menu bar.

Use Scrollbars ( ):

By default, the thumbnails are not scaled to fit the K-means window. If all thumbnails do not fit in the K-means window, scrollbars will appear to enable you to examine all thumbnails. If thumbnails have been scaled to fit the window, you can go back to using the scroll bars by clicking the button (**Use scrollbars**) or select **Thumbnails | Horisontal Scroll** from the K-means window menu bar. Set the thumb width by dragging the grey column-header.

Fit in Window ( ):

To scale the thumbnails according to the window size, click the button (**Fit in window**) or select **Thumbnails|Horisontal Fit** from the K-means window menu bar.

Visual Properties
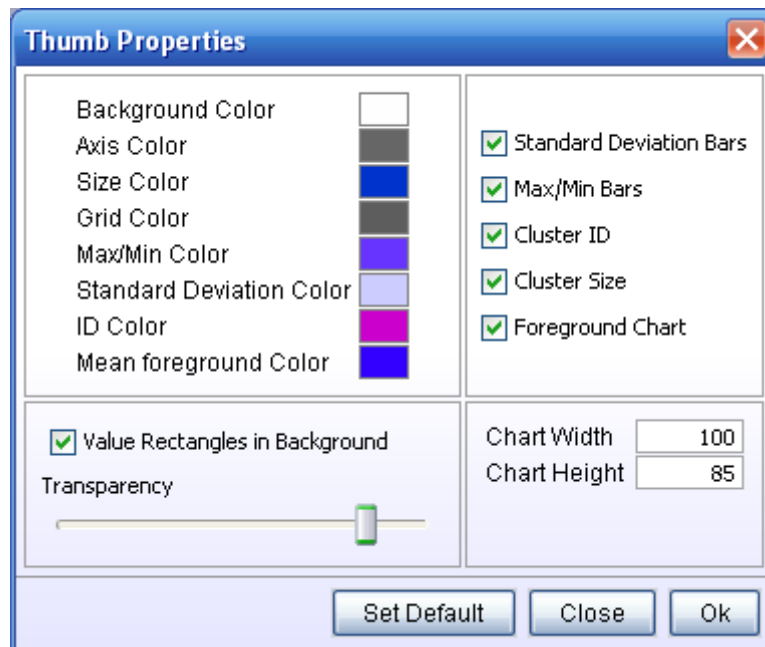
Right click on a thumbnail to set the visual properties for the K-means thumbnails, or select **Thumbnails | Thumbnail Layout** from the K-means window menu bar. The options are:

- Chart Width/Height - sets the width/height of the chart in pixels.
- Color options - click any of the colored boxes to set the desired color for that option.
- Paint Standard Deviation bars - check this box to include the bars indicating the standard deviation for each state.
- Paint Max/Min bars - check this box to include the bars that indicate the maximum and minimum values for each state.
- Include Cluster ID - check this box to display the cluster ID of a cluster on its thumbnail.
- Include Cluster Size - check this box to display the amount of profiles in a cluster on its thumbnail.

- Foreground Chart - check this box to display mean or all profiles in the thumbnail windows
- Value Rectangles in Background - if checked, it will display the values of the profiles in this cluster as colored rectangles, one row for each profile.
- Transparency - the slide bar only has an effect if the Value Rectangles in Background is checked. The slide decides the transparency of the foreground chart. Slide bar to the very left - 100% transparency - the value rectangles in the background shows strongly. Slide bar t the very right - 0% transparency - the value rectangles in the background cannot be seen.

Focusing on single clusters:

To focus on a single cluster simply click on its thumbnail. A new tab appears labeled with the ID number of that cluster. Clicking on the new tab brings up a Gene Graph viewer showing all the profiles contained in that cluster. Please refer to section 3.2 for more information about using the Gene Graph viewer.

Branch dataset (⌑):

One additional feature that exists for the zoomed selection in the K-Means window is to branch the dataset into a new node in the Project Tree. To do this, select the tab that contains the data you want to branch. Then click the ⌑ button on the K-Means window tool bar, or select **Line Chart | Branch Dataset** from the K-Means window menu bar. A new node will be added below the current one in the Project Tree labeled with the K-Means symbol ⠿.

Show Variance Diagram:

Select Thumbnails | Show Variance Diagram from the K-Means menu bar to open the Variance window. This window shows a square grid of cells, where each cell represents a cluster. The cells are color-coded to show the amount of variance in each cluster according to the color key table on the left side of the window. Hovering the mouse cursor over a cell displays a tool tip with the exact Single Variance, Between Variance and Cluster Size values of the cluster represented by the cell. If the box Clustersize as alpha is checked, the number of profiles in a cluster will indicated by the transparency of a cell against a gray grid background. A highly transparent cell contains few profiles, whereas an almost opaque cell contains many profiles. Click on a cell to highlight the corresponding thumbnail in the K-Means window. Click the Close button to close the variance window.

Remove tabs (⌑):

To remove a tab from the K-Means window select the tab to be removed. Then click the ⌑ button (**Delete Active Tab**) on the K-Means window tool bar, or select **Line Chart | Delete Active Tab** from the K-Means window menu bar.

Put in Tree

To place the entire component into the project tree click the ▣ button, **Thumbnails | Put in Tree** from the K-means window menu bar. This creates a new node with the symbol ▦ in the project tree that acts as a direct shortcut to the current component.

Additional buttons

The additional buttons that become active when selecting one of the tabs that appear when clicking on the thumbnails from the Clusters tab, are described in the Gene Graph section (3.2)

3.8 Principal Component Analysis
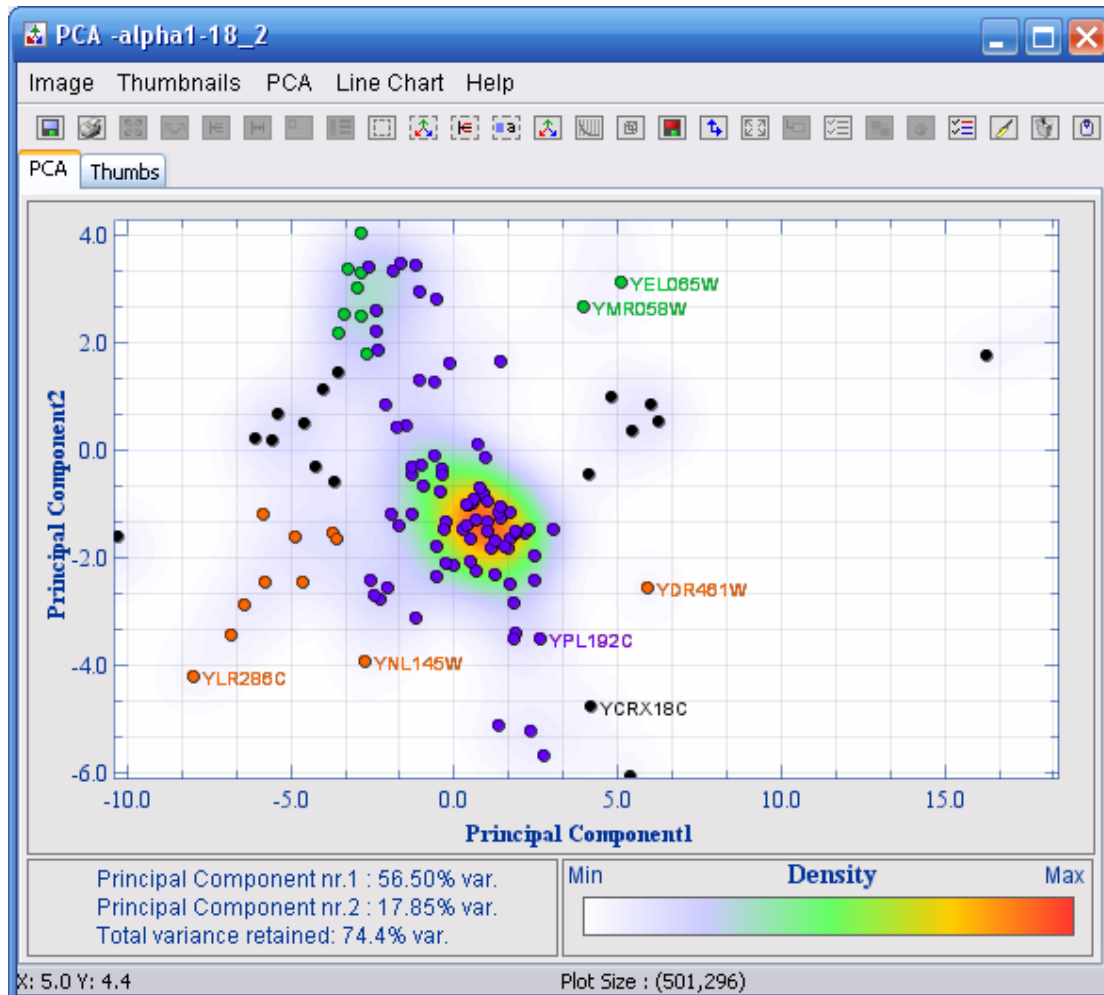
3.8.1 The PCA Window

Select the node you wish to run the analysis on in the Project Tree, and then click the

Principal Component Analysis (⬜ **)** button on the J-Express Pro tool bar. Alternatively, select **Methods | Principal Component Analysis** from the J-Express Pro menu bar.

The PCA window opens, and it follows the common pattern of most windows in J-Express Pro, with a menu and a tool bar, with an area below it for data display organized into tabs. When a PCA window is first opened it contains two tabs: PCA and Thumbs.

3.8.2 The PCA tab

The PCA tab shows a 2D plot of the dataset. The axis chosen by default are the ones that result in the highest total variance. Each profile is represented in the PCA plot as a dot. Additionally, the density of dots in each local area is indicated by a range of colors (by default white (lowest density) through blue and red to yellow (highest density)). Thus high numbers of dots in an area will be obvious, even though the dots more or less overlap. If a dataset is large, or the data is centered in a relative small area, it is possible to define a threshold value. If the dot density exceeds this value, the dots will be removed in this area. This makes it easier to see the underlying structure of the spread of the plot, and identify and select outliers. The variance of the axis and the total variance for the plot are displayed in the upper left corner. The color range for relative dot density is shown in the upper right corner of the plot.

To focus on an area of interest in the PCA plot, click the **Frame content to PCA(** 🔧 **)** and make sure that the **Frame method** is set to ⬚ **Square**. Drag out a selection box around the area by clicking and dragging with the mouse. The PCA plot will be zoomed to the selected area. Alternately you can select the **Frame contents to chart** button (📈). If the area contains any profiles, they will be added as a thumbnail to the Thumbs tab. Alternatively use the Lasso tool (✎) to draw the selection area. The lasso tool is found by clicking the **Frame Method** button and then selecting ✎ **Lasso**. Different types of fill can also be chosen for the selected area. Further PCA operation will only affect the selected area.

Customizing the PCA plot
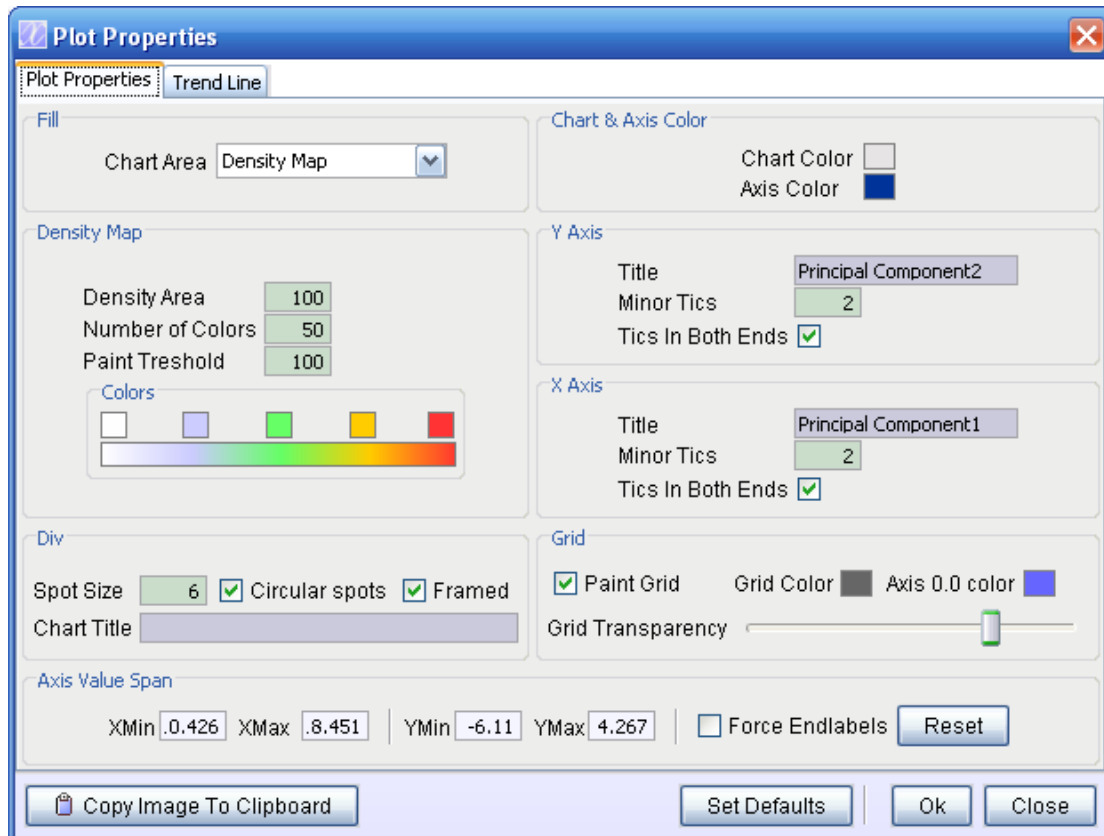
To customize a PCA plot, select **PCA|PCA Properties** from the PCA menu bar. Another way to bring up the PCA properties window is right clicking on the PCA plot

Fill lets you choose the background color of the PCA plot. The options are:

- One color – click the Background color box to select a new background color for the PCA plot.

- Density map – uses a spectrum of colors to show the density of PCA points in an area.

    Density Map options:

    These options become available when the density map is selected as the fill type.

    Density Map Colors – allows you to change the color of the highlights. To change a color in the PCA color range simply click one of the small boxes over the spectrum. This brings up a color selection dialog where you can choose the color you want. Click OK, and the color range will change to accommodate your changes.

    Density area – allows you to set the size of the area a single dot influences on the density map. To make the influence of a dot less, move the slider to the left, to increase the influence of a dot move the slider to the right.

    Number of Colors – sets the number of colors to be used to generate the density map. A smaller number of colors limit, and in some cases removes the density map for dots lying in areas of low density. In addition the transition between colors becomes less gradual. Move the slider to set the desired amount of colors to be used.

    Paint Threshold – sets a threshold value for the amount of dots in an area. If this threshold is exceeded the dots in that area are removed. This frequently helps show the structure of the Density Map. Move the slider to set the desired threshold.

- Gradient – Two colors are combined to create a smooth color gradient. Click the two colored boxes to choose the desired colors. Use the Gradient Type menu to select the type of gradient. Diagonal forms a color gradient from the upper left to the lower right corner; Top-Bottom forms a color gradient from the top of the plot to the bottom.

- External Picture – Use the file selection dialog to select the image file you wish to use as a background for the plot. Selecting Stretch will stretch the image to fit the plot. Selecting Tile will repeat the image in a tile pattern if it is too small to cover the entire plot.

- Tiles – Six additional patterns you can use for your plots.

Density Map

- **Density Area -** the value in the Density Area text field says how far out (in pixels) from a dot the density circle should stretch.
- **Number of colors -** the number of colors that will be used to draw the density areas.
- **Paint threshold (%) -** if certain areas are very dense, you may want to remove some of these profiles from the plot. This makes it easier to for instance spot differentially expressed genes. If the threshold value is set to i.e. 10, only the profiles belonging to the 10% least dense areas will be plotted.
- **Colors -** click on the colored squares to change the colors.

Div

- **Spot size** lets you set the size in pixels of the PCA points.
- **Circular Spots** - check this box to use circular PCA points.
- **Framed** - Checking this box adds a frame around each dot.
- **Title** - enter a title for your chart in this box, if needed. It will appear at the top of the chart.

Axis Value Span lets you set the maximum and minimum values for each axis. Uncheck the Force Endlabels box to turn off the automatic endlabels generated by J-Express Pro. Click the Reset button to reset the value span.

Chart & Axis color - click these colored boxes to set the background color for the area outside the main chart, and the colors used for the axis.

X- and Y-axis options

- Title allows you to name each axis. The name will appear on the left side of the chart for the y-axis, and on the bottom of the chart for the x-axis.
- Minor tics sets the amount of minor tics between each major tic on the respective axis.
- Tics on both ends - check this box to have tics on the opposite edge of the plot from the axis, in addition to the tics on the axis.

Grid lets you set options for the plot grid.

- Paint Grid - check this box to toggle display of the grid on. Uncheck it to toggle display of the grid off.
- Grid Color - select the desired color for the grid by clicking on this box and choosing a color from the dialog that appears.
- Grid Transparency - Use this slider to set the transparency of the grid, relative to the background.
- Axis 0.0 color - click on the colored square to change the color of the X and Y axis (i.e. X=0.0 and Y=0.0)

All changes made in the PCA properties window take effect as soon as you click OK. To set the current settings as default click the **Set Defaults** button.

Additional PCA tab features

Save images (▣):

To save an image of the PCA plot, click the ▣ button on the PCA window tool bar. Select the location and name of the file and click **Ok**.

Printing (▣):

To print the PCA plot click the ▣ button on the PCA window tool bar.

3D PCA Scatter Plots (▣):

To see the entire plot in three dimensions click the ▣ button on the PCA window tool bar, or select **PCA | Create 3D PCA Scatter Plot** from the PCA window menu bar. This creates another tab in the PCA window marked "3D". If you click on this tab you will see a 3-dimensional model of the scatter plot. Only the dots are shown. To rotate the model, click the ▣ button (**Rotate 3D Scatter Plot**) and then click and drag in the window. To zoom in or out on the model click the ▣ button (**Zoom 3D Scatter Plot**) and then click and drag in the window.

Save Projection and Eigenvalues (▣):

It is possible to save the projection and eigenvalues of the PCA plot to a tab-delimited file. To do so, click the ▣ button (**Save Projection and Eigenvalues**) on the PCA window tool bar, or select **PCA | Save Projection and Eigenvalues** from the PCA window menu bar, and then choose a location and a file name in the dialog that appears. The first line of the file lists the eigenvalues. The next line lists the headers (if any are available) for the columns. Then follows the projections for each profile, using 1 line for each profile. Information in the defined info areas is included if available.

Show Principal Components (▣):

To view all the principal components of the dataset click the ▣ button (**Show Principal Components**) on the PCA window tool bar, or select **PCA | Show Principal Components** from the PCA window menu bar. This opens a Gene Graph window showing all the principal components. For more information on using Gene Graph windows and functions, please refer to Section 3.3.7.

Principal Component Variance (▣):

To view the variance of the principal components (the eigenvalues) click the ▣ button (**Principal Component Variance**) on the PCA window tool bar, or select **PCA | Principal Component Variance** from the PCA window menu bar. This brings up a Gene Graph window showing the principal component variance. For more information on using Gene Graph windows and functions, please refer to Section 3.3.7.

The Thumbs tab

Whenever a selection rectangle is defined that covers one or more dots (profiles) on the PCA plot, a new thumbnail is created on the Thumbs tab, containing the profiles selected.

The Thumbs tab has the same functionality as the K-Means thumbnails. For additional information see section 3.7.1

Deleting a tab ( ):

To remove a tab from the PCA window, select the tab to be removed. Then click the button (**Delete Active Tab**) on the PCA window tool bar, or select **Line Chart | Delete Active Tab** from the PCA window menu bar. To remove the 3D scatter plot from the menu bar select **PCA | Delete Active Tab** instead. The PCA and Graphs tabs can't be deleted.

Branch dataset ( ):

One additional feature that exists for the zoomed selection in the PCA window is to branch the dataset into a new node in the Project Tree. To do this, select the tab that contains the data you want to branch. Then click the button on the PCA window tool bar, or select **Line Chart | Branch Dataset** from the PCA window menu bar. A new node will be added below the current one in the Project Tree labeled with the PCA symbol .

Choose Axis ( ):

In the 2D and 3D pca plots, the axis representing the 2 and 3 greatest variances respectively, are selected as default. To view the plots using other axis, press the **Choose Axis ( )** button or select **PCA | Set Chart Axis**, and select the axis you want to use from the pull down menus.

Show Location Thumbs

To get an instant thumbnail of the profile represented by a PCA point, select **PCA | Show Location Thumb**. This will bring up a small thumbnail window, which will show a thumbnail of the profile represented by the point the mouse cursor is currently over. This window has the same functionality as the Project Thumbnail window.

Show Variance:

Checking/un-checking **PCA | Show Variance** toggles display of variation statistics on or off.

Show Density Scale:

Checking/un-checking **PCA | Show Density Scale** toggles display of the Density Scale on or off, if the density map is being used.

Show tool tip box:

To get any available additional information defined in information columns of the data shown as tool tip text check the **PCA | Show tool tip** box. When the mouse pointer is held over a PCA point, the additional information (if any) will be shown next to it as a tool tip.

Zoom (⬚):

To zoom in on an area of interest on the PCA plot, click the **Frame Method** button, and then drag out a selection box. The PCA window will zoom in on the selected area. To zoom back out, click the **zoom out** ⬚ button. Note that zooming only works with the square selection tool (⬚).

The Three next choices are complementary. By selecting one way of handling framing of spots, you disable the two other.

Frame Contents to PCA (⬚):

This option sets the zoom flag so that all framing (with square) are zoomed.

Frame Contents to Chart (⬚):

This option sets the chart flag so that all spots being framed are put into a thumb diagram. This feature lets you fish out interesting areas with spots and view the corresponding elements profiles.

Toggle labels on FrameContents(⬚):

This option lets you select the spots to have labels. After clicking on this button, you can either click on each spot you want labelled, or drag a lasso or frame over multiple spots.

Shadow unselected (⬚):

To select certain genes, frame the area containing the genes you want to selected to chart (click ⬚ and drag out a selection box). Select the Thumbs tab and click on the new thumb. This will open a gene graph window. Genes can be selected from the list displayed to the left. Click the **Shadow unselected (⬚)** button. The selected genes will now be clear while the unselected genes will have a shade of grey. If you go back to the PCA tab and click the **Shadow unselected (⬚)** button once more, the selected genes will be clear, while the others will have a shade of grey. If other genes are selected, the clear and shadowed genes are updated automatically. To un-shadow unselected, simply click the **Shadow unselected (⬚)** button again.

Repaint Component (⬚):

If changes you make do not take effect immediately, press the **repaint (⬚)** button.

Copy Clip Image to Clipboard (⬚):

To copy the image in any of the tabs to clipboard, click the ⬚ button.

Initiate K-Means

You can do K-Means clustering of the entire dataset based on the mean of the defined thumbnails. To this, create one or more PCA thumbnails by clicking the **Frame contents**

to chart button () and dragging out selection areas. Then select **Thumbnails | Initiate K-Means**. This will start K-Means analysis on the entire dataset using the mean of the thumbnails as the initialization method, and the number of cluster equal to the number of thumbnails in the PCA window.

Put in Tree

To place the entire component into the project tree click the button, **PCA | Put in Tree** from the PCA menu bar. This creates a new node in the project tree that acts as a direct shortcut to the current component.

3.9 The Self Organizing Map
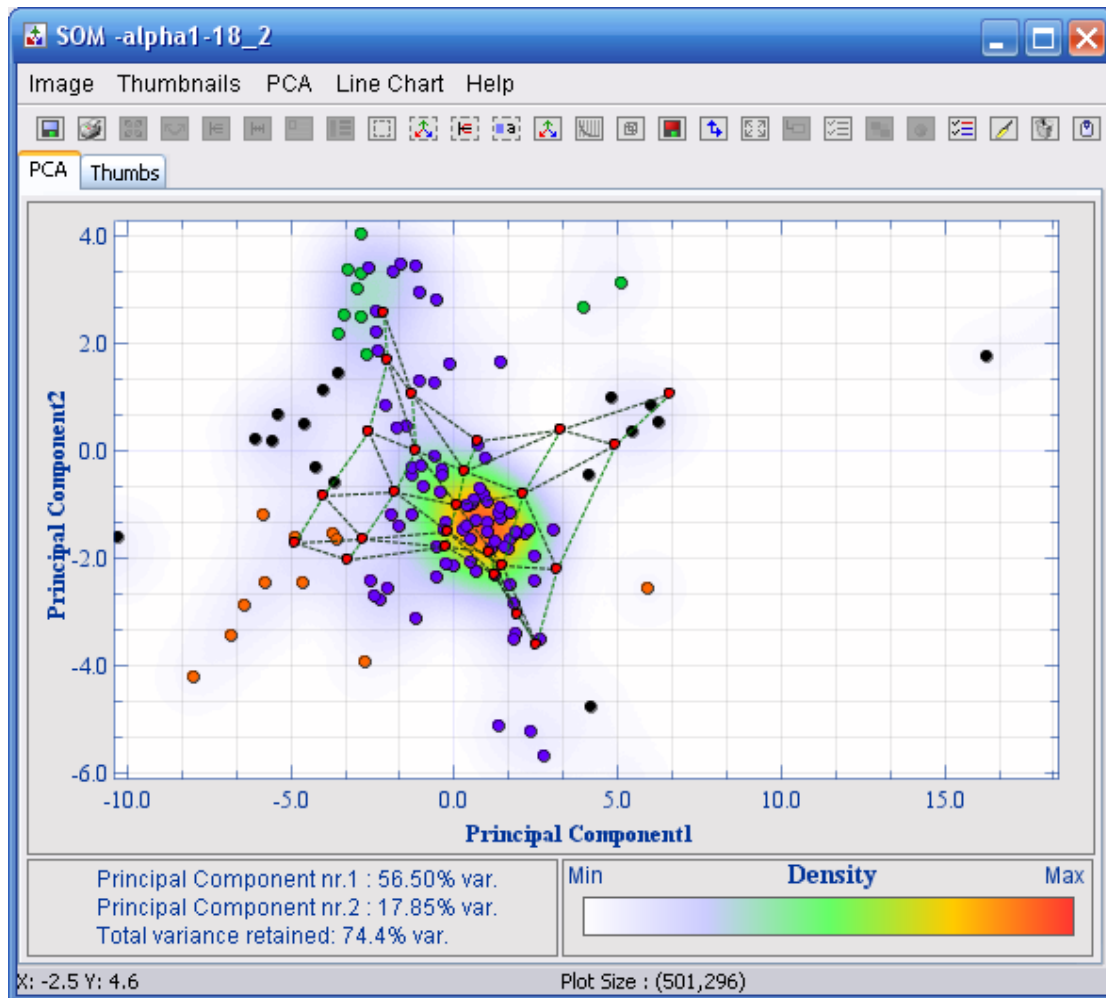
To open the Self Organizing Map (SOM) window select the node in the Project Tree for analysis, and click the ⊞ button (**Self Organizing Map**) on the J-Express Pro tool bar. Alternatively select **Methods | Self Organizing Map** from the J-Express Pro menu bar. The SOM properties dialog will open.

There are two ways of executing the SOM in J-Express. The default easy way is to set all parameters automatically and just select the number of neurons (clusters).

The other way is to select *advanced* and set all the parameters manually. In the advanced tab you can select the *Visualize in PCA Window* option and view how the neurons adapt to the underlying data in a PCA window. The parameters for the advanced option is described below.
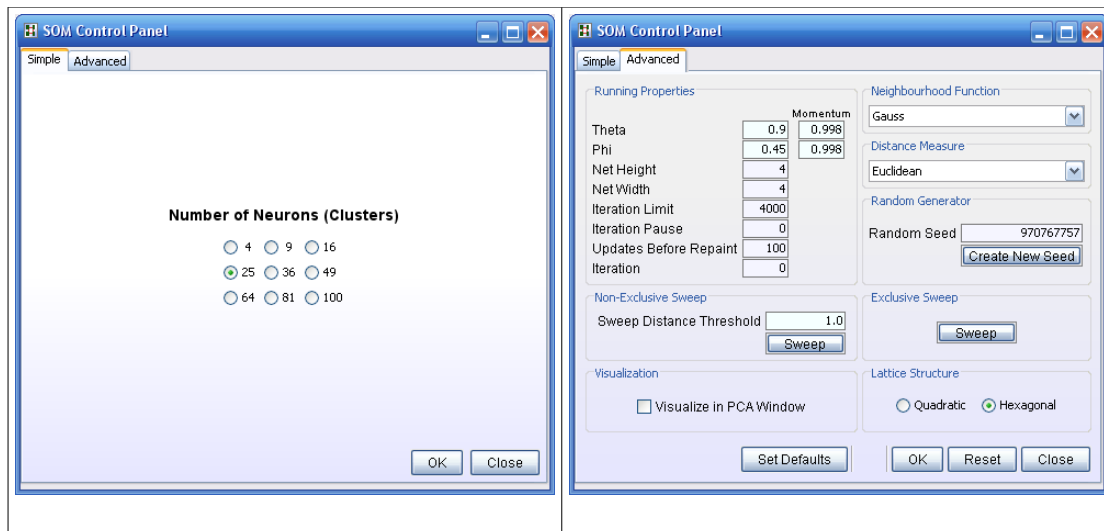
The SOM properties window.

Parameters in the SOM properties window

Running Properties:

> Theta/Momentum – This affects the initial distance a neuron is moved towards a data point when the map is adapted to fit the data set in the training phase of the SOM. The Momentum box gives an opportunity to set the "friction"-rate when moving a neuron. The Momentum is constant during the training phase.

> Phi/Momentum – This sets the amount of "pull" working between neurons, in other words how much the neurons should affect each other. The Momentum box sets the "stiffness" of the links between neurons. The Momentum is constant during the training phase.

> Net Height – The number of horizontal neurons in the neuron lattice.

> Net Width - The number of vertical neurons in the neuron lattice.

> Iteration Limit - sets the number of iterations the training algorithm in the SOM should perform. The algorithm can be stopped and restarted before the iteration limit value is reached, by clicking the Stop/Run button.

> Iteration Pause – sets the amount of time the algorithm should pause between iterations. Use iteration pause if you want to follow the training of the SOM from iteration to iteration visually.

> Updates before Repaint – sets the number of iterations that should be performed before the graphical view of the SOM is updated.

> Iteration – shows the number of iterations the SOM algorithm has performed.

Neighborhood Function:

Use the pull-down menu to select which neighborhood function you wish to use for the neurons. For information about the different neighborhood functions please refer to Section 4.4.2.

Random Seed:

The Random seed is used as a basis for the randomizing the algorithm needs. If you need to repeat a particular analysis, enter the same random seed and keep all other options the same to get the same result. A random seed number can be any (large) number. Click **Create Random Seed** to generate one automatically.

Distance Measure:

To choose a different distance measuring method, choose a new one from the **Distance Measure** list. For definitions of the different distance measures, please refer to Section 4.1.

Sweep Distance Threshold:

The Sweep circumference is used as a parameter for the Sweep and Exclusive Sweep functions. It sets a distance where points lying within this distance should be included in the sweep.

Lattice Structure:

With this option you can choose between quadratic or hexagonal neuron lattice structure.

Visualization:

Check the **Visualize in PCA window** box to show the SOM as an overlay on a PCA window. Not visualizing the SOM analysis can be useful if you're only interested in performing sweep operations, since the analysis will be somewhat faster.

3.9.1 The Self Organizing Map

If **Visualize in PCA window** is selected, the SOM is shown superimposed over a normal PCA window. The coordinates of the neurons and the data points are shown. The coordinates are defined by the two first principal components calculated for the data set. For information about the PCA window, please refer to Section 3.6. Each neuron is shown as a red dot, with green lines connecting it to its neighboring neurons. As the algorithm proceeds the neurons will be moved around, in an attempt to fit the neurons to the data set. To run the SOM algorithm again, click the Reset button followed by the start button. To continue with an existing map (e.g. after some of the parameters are changed) input a new (larger) value in the **Iterations** box of the SOM properties window, and click **Run**.

Remember that the visualization shows a reduced representation of the data points and the neuron network (since only the first two principal components are used). The Self Organizing Map will show up on the 3D scatter plot, and is useful in situations where the SOM seems to collapse. In these situations the SOM is fitting itself to the data in a way that the 2D PCA data window cannot display properly. By viewing the SOM in the 3D scatter plot, one more principal component is used to define the coordinates, and more

information is preserved in the view. In most cases, the 3D visualization will show more information than the 2D, but not all the information in the data set will be shown.

3.9.2 Operations on SOMs

All operations that function on PCA plots work in the SOM window. Two operations unique to the SOM are the sweep and exclusive sweep.

Sweep and Exclusive Sweep

To perform a sweep operation, click the **Sweep** or **Exclusive Sweep** button on the SOM properties window. A new tab is added to the SOM window. The tab contains one thumbnail for each neuron in the map. Each thumbnail contains the mean profile of all profiles lying within the sweep circumference (set in the SOM Properties) range of that neuron. When an exclusive sweep is performed, a data point is assigned only to the closest neuron. The new tabs are labeled as "Sweep 1", "Sweep 2" etc. in the order they were created.

These thumbnail tabs have the same functionality as the K-means Thumbs window (see Section 3.7.1). The labels for the focused thumbnails are different however. A focused thumbnail will be labeled like this: SW 1 Cl. 1 indicating the tab contains the results of sweep number 1, with the focus on the 1st neuron of the SW1 tab. The neuron represented by the last thumbnail focused on is highlighted on the PCA plot with a small blue box.

3.10 The spreadsheet

To reopen the spreadsheet used when importing data into J-Express Pro click the ▦ button (**Tabular View**) on the J-Express Pro tool bar, or select **Methods | Spreadsheet** from the J-Express Pro menu bar. Here you can correct erroneous data, enter new data manually, or paste in spreadsheet information from other applications, (e.g. Microsoft Excel ™).

3.11 Similarity search
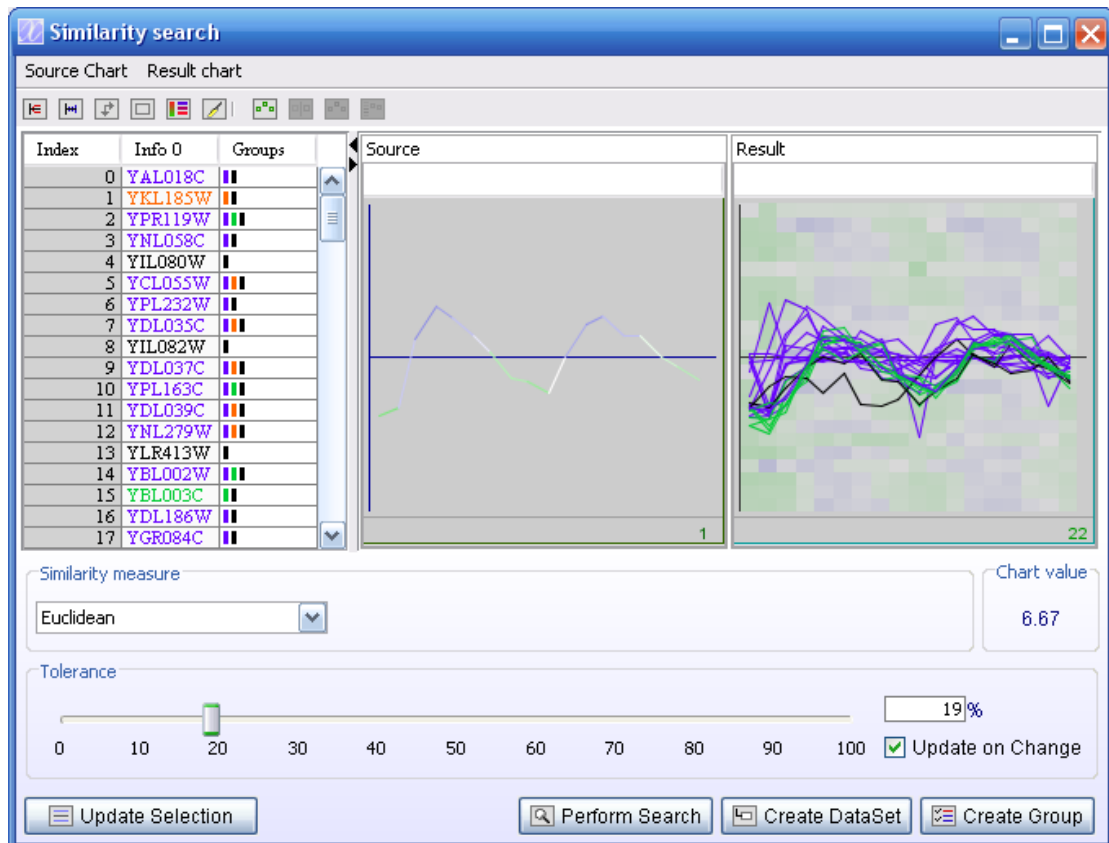
Often during data analysis certain profiles seem to follow a similar pattern. J-Express Provides the Similarity search method as a tool to find all profiles within a certain range of similarity. The similarity between profiles can be calculated by a variety of different distance measure schemes, allowing the user maximum flexibility in detecting common patterns in the dataset. It is also possible to build a profile from scratch and search your dataset for similar profiles.

Select the node you want to analyze by clicking on it in the Project Tree. Then press the ↔ button (**Similarity Search**) on the J-Express Pro tool bar, or select **Methods|Find Most Similar** on the J-Express Pro menu bar.

Click the **Show all profiles** button (▣) to display all the profiles in a thumbnail, rather than a mean profile. To go back to showing the mean profile only click the **Mean Profile Only** button (▣).

The profile thumbs in the window are scaled to fit the window size by default. To disable this scaling and use scrollbars to see parts of a profile outside the visible area of the window, click the **Use Scrollbars** button (▣). To go back to scaling the profiles automatically to the window size click the **Fit in Window** button (▣).

Enable/disable the use of group colors on the thumbnails by clicking the **Toggle group colors** button (▣).

Update On Change

Check this box to update the visualization of the Find Similar Profiles as you drag the slider defining the proportion of the closest expression profiles to be displayed. If you are analyzing a large data set, the interactive updating may become slow, in which case this check box should be de-selected.

Charts

The Charts area contains two thumbnails, Source and Result. The Source thumbnail contains a preview of the selected profiles. The Result thumbnail shows the profiles that lie within the selected range of similarity.

If multiple profiles are selected, the Source thumbnail will display the mean profile. Right click on a thumbnail to set the visual properties for it. For more information on setting thumbnail properties see Section 3.7.1, visual properties.

Both thumbnails can be clicked to open a Gene Graph viewer displaying the full profiles. For more information on the Gene Graph viewer please refer to Section 3.3.7.

Distance Measure

To choose a different distance measuring method, choose a new one from the **Distance Measure** list. For definitions of the different distance measures, please refer to Section 5.1. This tool provides an instructive way to study the difference in behavior between the different distance measures.

Tolerance (%)

The tolerance slider allows you to set the amount (in per cent) of similarity that is needed for the profile to be included in the search. Move the slider to set a new percentage value.

Update Selection

The ▤ **Update Selection** button makes a selection of the selected profiles and the profiles within the similarity tolerance take effect in all windows in J-Express Pro. If you for instanse have a Gene Graph window open simultaneously, the selected profiles can be highlighted using the shadow unselected feature.

Create Dataset adds a new branch to the Project Tree below the current one containing the profiles that were returned by the search.

Create Group adds a new group to the Manage Groups window named "Closest Set". The group can then be used as any other in J-Express Pro.

3.11.1 Create Profile

To create a profile from scratch and use this to search for similar profiles in your dataset. Click the **Create Profile (** ⬚ **)** button. This enables three other buttons which will be described below. To go back to searching for profiles from the list, click the **Use Mean Of Selection As Source (** ⬚ **)** button.
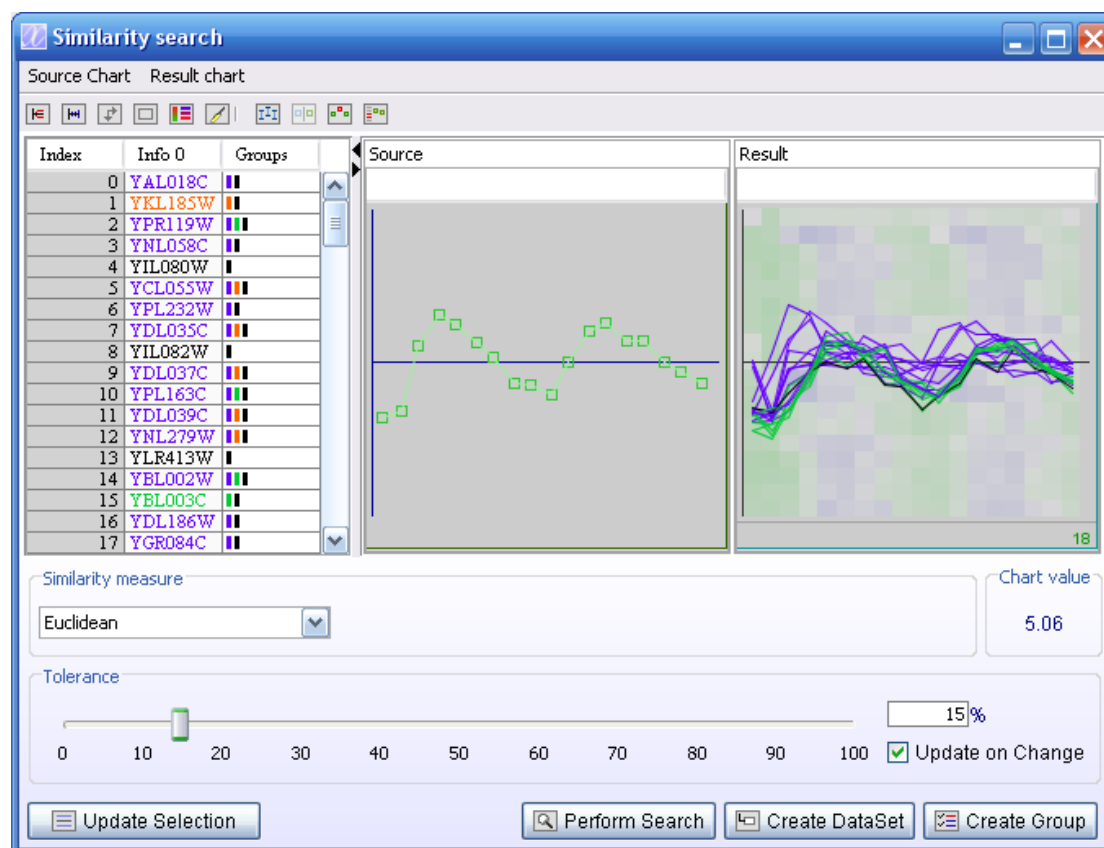
In the source plot area, there is a line of green boxes along the X-axis at Y=0. Each green box represents a column in the dataset. To create your profile, simply move each box up or down to the wanted location.

Select The Columns (Not) To Use In The Distance Calculation (⬚ **):**

If you only wish to use certain columns for your created profile, click the ⬚ button. Next click and drag the mouse to deselect the columns you do not wish to use. The deselected columns gets a blue color.

Select Columns To Change (⬚ **):**

To create your profile you want to move the columns up or down. Click the **Select Columns To Change (** ▥ **)** button. Next click and drag the mouse to select the columns you wish to move. The selected columns get a red color. To move the columns click in one of the red squares and drag to wanted location.

🔍 **Perform Search**

To search the dataset for your created profile, click the 🔍 **Perform Search** button. The result is displayed in the **Result plot area**. Keep in mind that the number of profiles you get back depends on the **Tolerance**.

Create Source Profile From Selected Mean (▥ **):**

The **Create Source Profile From Selected Mean (** ▥ **)** button allows you to use the mean of the genes selected in the **Table Columns** as a starting point for your profile design.

Close - press this button to close the Find Similar Profiles window.

3.12 Profile Search
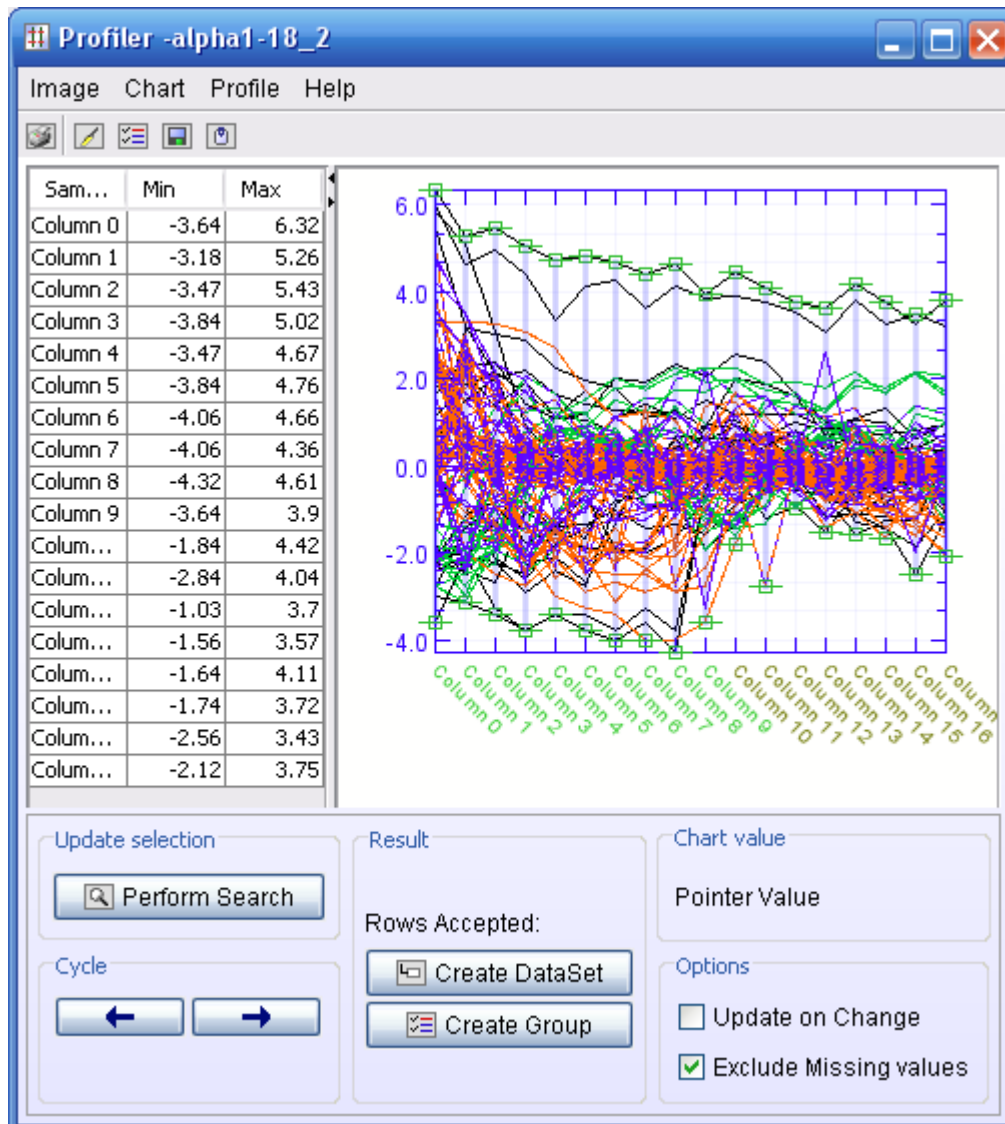
The Profiler allows you to specify boundary profiles that are used as a basis for finding existing profiles in the dataset.

Select the node from the project tree that you want to analyze, and press the **Profile Search (** **)** button on the J-Express Pro tool bar.

3.12.1 Profile design

The profile design area displays a thumb view of all the profiles in the dataset. For each state, there are two green boxes; at the lowermost and uppermost profiles

respectively. To search for profiles that have values between a smaller range for a particular state, move the boxes that mark the particular state to the new maximum and minimum values for your search. The search will return the set of profiles whose expression values are between the minimum and maximum values (defined by the lowermost and the uppermost profiles) for each of the states.

To **move a green box**, click on it so it becomes red, click and drag the red box to the new location. To value at the mouse pointer is updated and printed behind the **pointer value** tag, every time you move the mouse in the thumbs area. To **unselect a red box**, simply click anywhere else in the thumb view window, but a box. This will turn all boxes back to green.

If several boxes are marked red, moving one of them will also move the other red boxes by the same amount.

3.12.2 Update On Change

Check this box to perform a new search every time a change is made to the search profiles. The number of profiles returned by the search will be updated behind the **Rows Accepted** tag.

3.12.3 Cycle

The Cycle (← →) left or right buttons will shift the values of the **Search Profiles** one state to the left or right respectively.

3.12.4 Perform Search

Click the Perform Search button (🔍)to find the profiles that lie within the bounds of the search profiles. The thumbnails window will be updated with the found profiles, if any exist. The number of profiles returned by the search will be printed behind the **Rows Accepted** tag.

3.12.5 Create Dataset

This button (🖳) adds a new branch to the Project Tree below the current one containing the profiles that were returned by the search.

3.12.6 Create Group

This button (📧) adds a new group to the Manage Groups window named "Profiler". The group can then be used as any other in J-Express Pro. See Section 3.1.14 and 3.1.15 for information on Creating and Managing groups.

3.12.7 Repaint Component

If changes you make do not take effect immediately, press the **Repaint Component (🖊)** button, or select **Chart | Update Chart**.

3.12.8 Additional Profiler Features:

🖼 Saving a Profile

To save a profile you have created select **Profile | Save Profile** from the Profiler window menu bar. Enter the file name you want (with .prf extension) and choose a location for the file in the dialog that appears. Click **Ok**, and the profile will be saved to disk.

🖼 Loading a Profile

To load a profile from disk, select **Profile | Load Profile** from the Profiler window menu bar. Locate the file containing the profile you want to open in the dialog that appears, and click Ok. The profile will be loaded into the Profiler window, replacing any existing content.

New Profile

To start a completely new profile, select **Profile | New Profile**. The contents of the Profile design areas will be reset.

Save an image

To save an image of the search profiles, click the 🖼 button on the Profiler window tool bar, or select **Image | Save** from the Profiler window menu bar. Select the location, name and appropriate format for the file, and click Ok.

Printing search profiles

To print the search profiles click the 🖼 button on the Profiler window tool bar, or select **Image | Save** from the Profiler window menu bar.

Setting Plot options

Right clicking on the graph or selecting **Chart | 🖼 Chart Layout** will bring up the Plot Properties dialog. Here you can alter most visual aspects of the Profiler.

Copy Image to Clipboard (🖼)

To copy an image to clipboard, press the **Copy Image to Clipboard (🖼)**.

3.13 Pathway Analysis
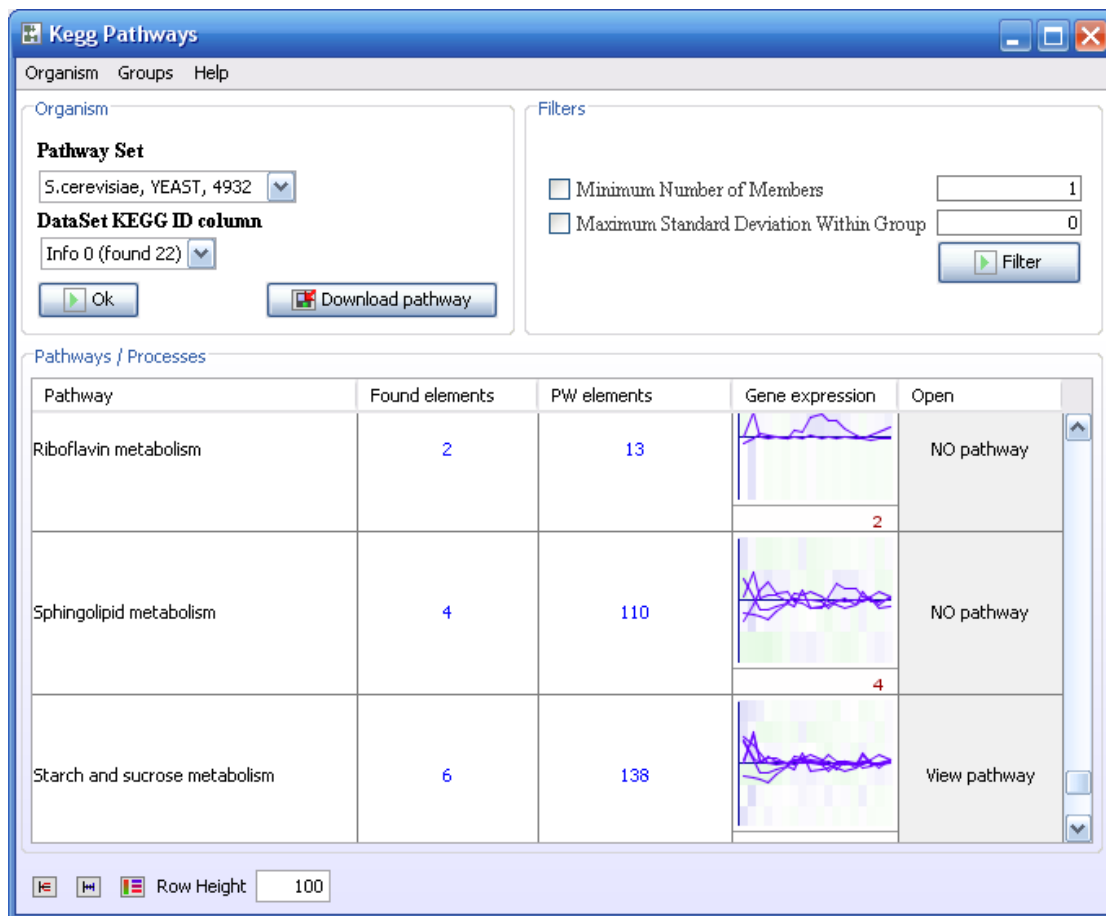
The Pathway Analysis component can be used to find clusters of co-expressed genes sharing the same pathway. This can give you an idea about why they are co-expressed.

Pathway Set

Select the correct **Pathway Set** for your dataset. If you cannot find the right pathwayset, you can download it by clicking on the **Download Set** button.
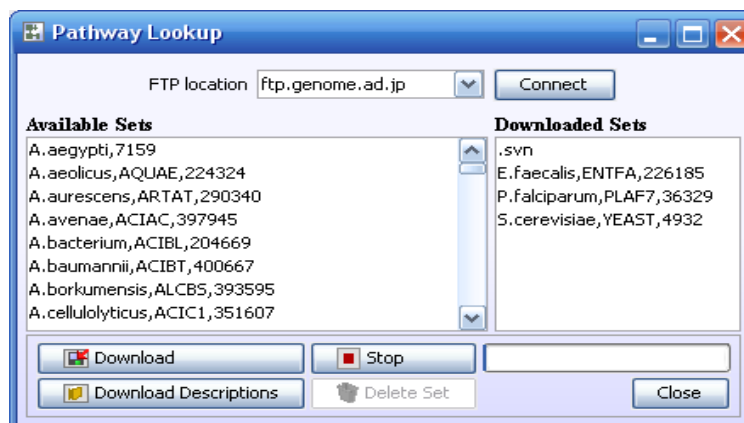
Select the correct organism and click download. This will download the KEGG pathway data and put it in the J-Express Pro resources/PW folder.

Download Descriptions

Clicking this button will download a file called map_title.tab to the J-Express Pro resources/PW folder. This file contains the kegg pathway id and its pathway name.

DataSet Locus Link ID column

Select the column from your selected dataset that contains the KEGG id's. For some organisms this will be the column containing the systematic gene names, while for others the KEGG id's will have to be downloaded from http://www.kegg.com and linked to the dataset using the J-Express Pro ID-Linker. This can be found under **Methods| IDLinker** on the J-Express Pro menubar. See IDLinker for details.

Filters

If you only want to analyse the pathways that have at least a minimum number of genes associated with it, check the **minimum number of members** box and enter the a number in the text field provided. Click **Filter**. This will remove all pathway entries with less members than the specified number.

If you are looking for pathways that have genes with similar expression profiles, you can filter on **maximum standard deviation within group**. Check the box and enter the maximum standard deviation. Click **Filter**. This will remove all pathway entries with higher standard deviation than the specified number.

The lower half of the window displays some clickable boxes.

- **Select** - To create groups for the different pathways, check the boxes in the Select column. Next press the **Create Groups from Selected** button. This will add one group for each selected pathway to the J-Express Pro Group Controller.
- **Name** - Displays the name of a pathway. Open a gene graph window from the J-Express Pro main menu bar (labelled: Line Chart). Press the Shadow Unselected( ) button. Arrange the windows so that you see both the Kegg Pathways window and the gene graph window. Clicking the pathways in the Name column will display the profile of the genes belonging to this pathway in the gene graph window.
- **Group** - The graph in these boxes have the same properties as the K-means thumbs. You can toggle display of all or mean profiles by clicking the and buttons respectively (lower left hand corner). The red number in the left hand corner of the group thumb shows how many members this particular group has. The green right hand number shows the group number. Clicking the thumb will open a gene graph window displaying the genes from this group. The button will open a thumbs properties window. If the button is clicked so that all profiles are painted, clicking the button will only paint the profiles that are members of a group in its group color.

- **Open Pathway -** Some of the boxes in the last column displays the text **Open Pathway.** The ones that displays this text can be clicked, and a window showing the molecular components will be opened:
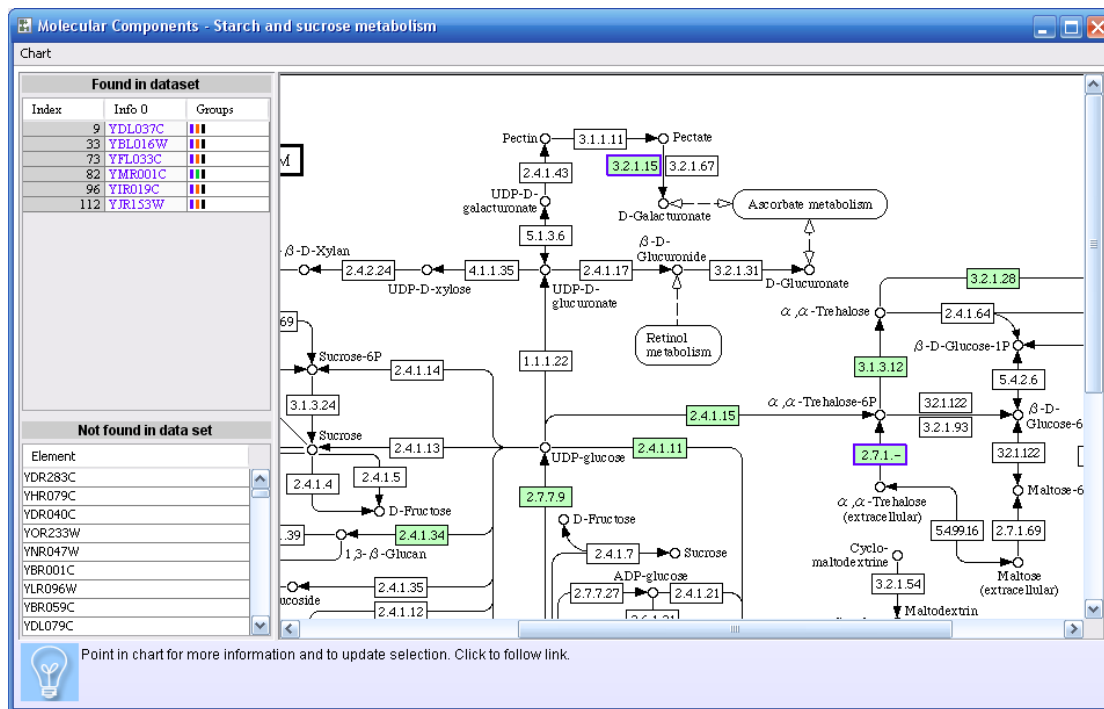
The list to the left of the divider displays the genes from the selected dataset that are members of this pathway. Highligthing the genes will also highlight these genes in the diagram to the right of the divider (multiple rows can be selected). If you look at the gene graph window the profiles of these genes will be highlighted there as well (as long as the Shadow Unselected(🗹) has been clicked).

Pointing the mouse on a circle or a frame in the pathway diagram will give some of them a yellow edge. When yellow click the mouse, and you will be taken to an external database for that particular component. For more information on the pathway diagram, see the KEGG site, http://www.kegg.com

Score Groups

The **score groups** feature is useful if you have a group created during earlier analysis, and you want to see if there is some statistical relation between that group and one of your pathway groups.

Two different statistical tests can be performed; Chi Square and Fisher-Irwin. It is also possible to set a limit to only include values that are statistical significant. The default value of 3.84 is found in a significance table and is the value corresponding to a 95% confidence limit. Click **Update.**

The Group Names are found in the first column, followed by the croup chart. The following charts show the profiles of the hits between a group and a pathway, and the number of hits are printed in blue. Underneath each chart is printed x2: (Chi Square) or FI: (Fisher-Irwin), and the result from the test. For more information on Chi Square and Fisher-Irwin, see any statistics text book.

Manually Downloading and installing pathways

You can download or update pathways manually from for instance the KEGG database. Locate the folder called resources/PW under the J-Express installation folder. Each organism has its own folder with the files

<ORG>_gene_map.tab
<ORG>pfa_synonym

and a collection of gif and conf files. These files can be copied from other locations, such as ftp://ftp.genome.ad.jp/pub/kegg/pathway/organisms but rememeber to put the new folder under the resources/PW folder.

3.14 Cell cycle analysis
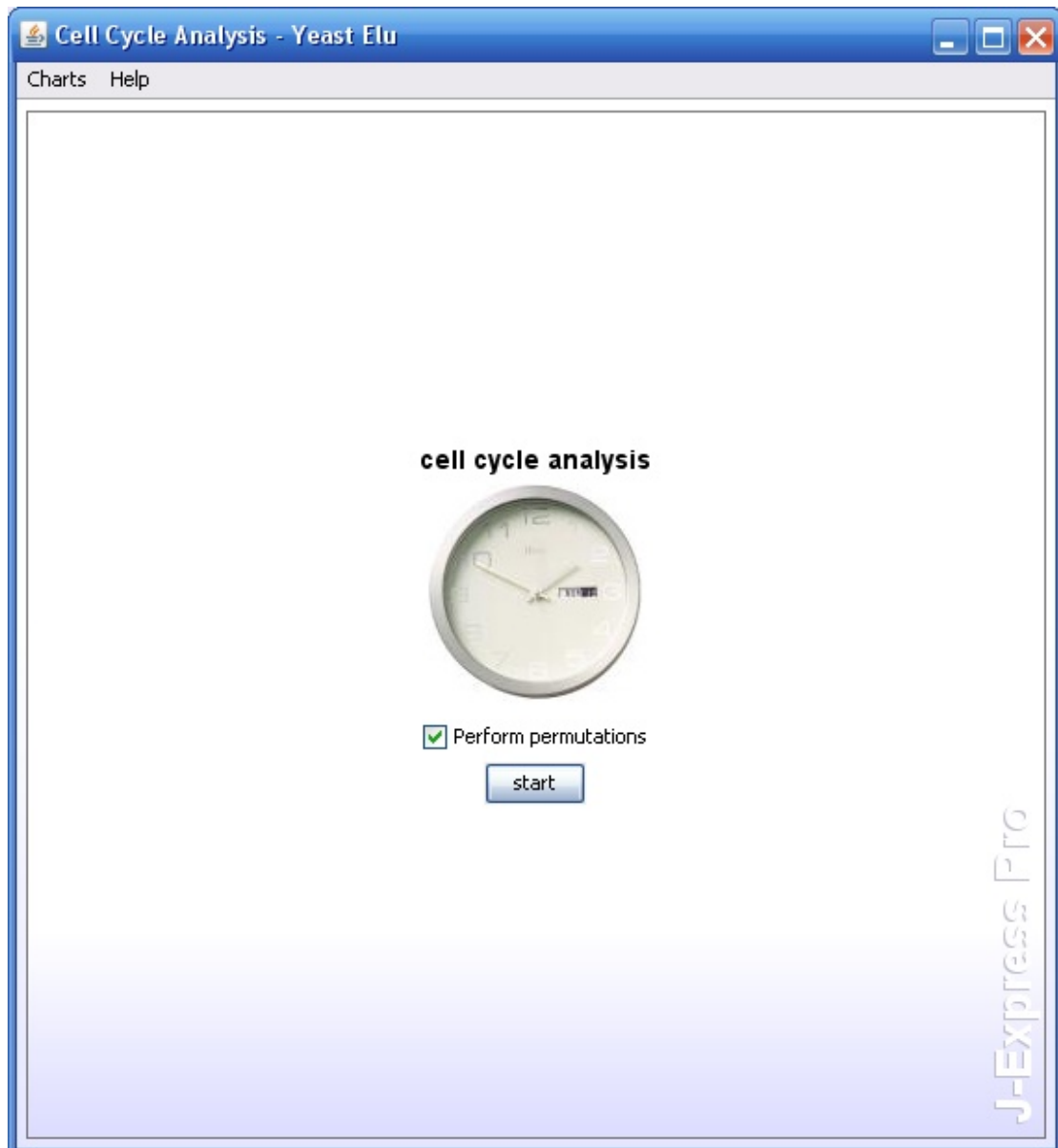
Use this component to find genes resembling a periodic regulation. The phase chart lets you select the number of phases in your experiment. For instance, if the experiment are arrays from samples taken every hour for one day, you have one phase. If you sample for two days, you have two phases etc. The green curve is the sine function and the blue is the cosine function. By pressing the "Auto Set" button, J-Express will predict the best phase settings, but **prediction will assume that the time between each sample is equal**. If this is not the case, you will have to use the "Edit" button to manually lay out the samples across the phase setting.
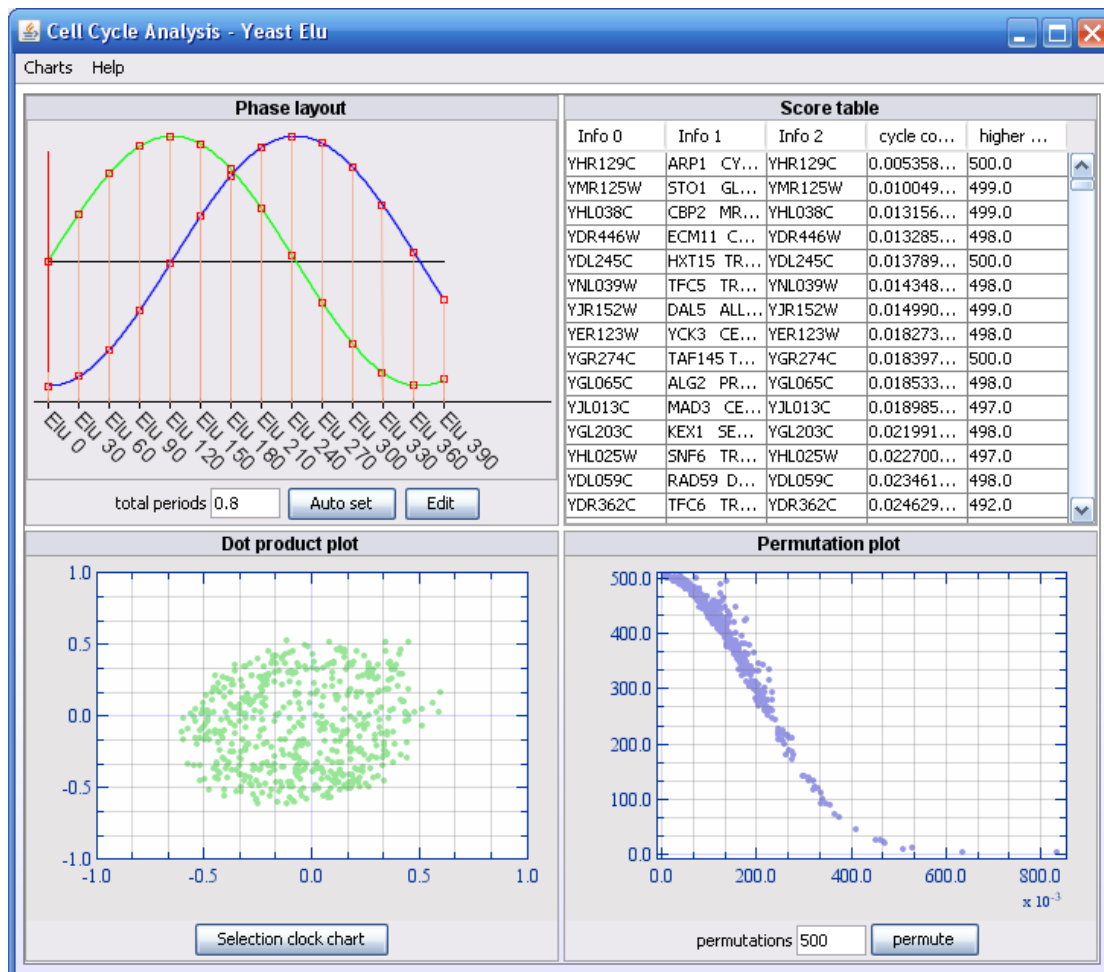
The dot product plot shows the rows in your dataset projected on the sine and cosine vectors defined in the phase component. In short, genes showing low or none periodic patterns will be located in the center of the plot while periodic genes will be located further from the origo. The x-axis in this plot is the sine function and the y-axis is the cosine function.

The score table sorts the genes according to the permutations performed and is linked to the permutation plot. This displays "Cycle correlation" as

Math.sqrt(sineprojection^2 +cosineprojection^2)

and "Higher permuted correlation" as number of permuted Cycle correlations above the unpermuted gene.
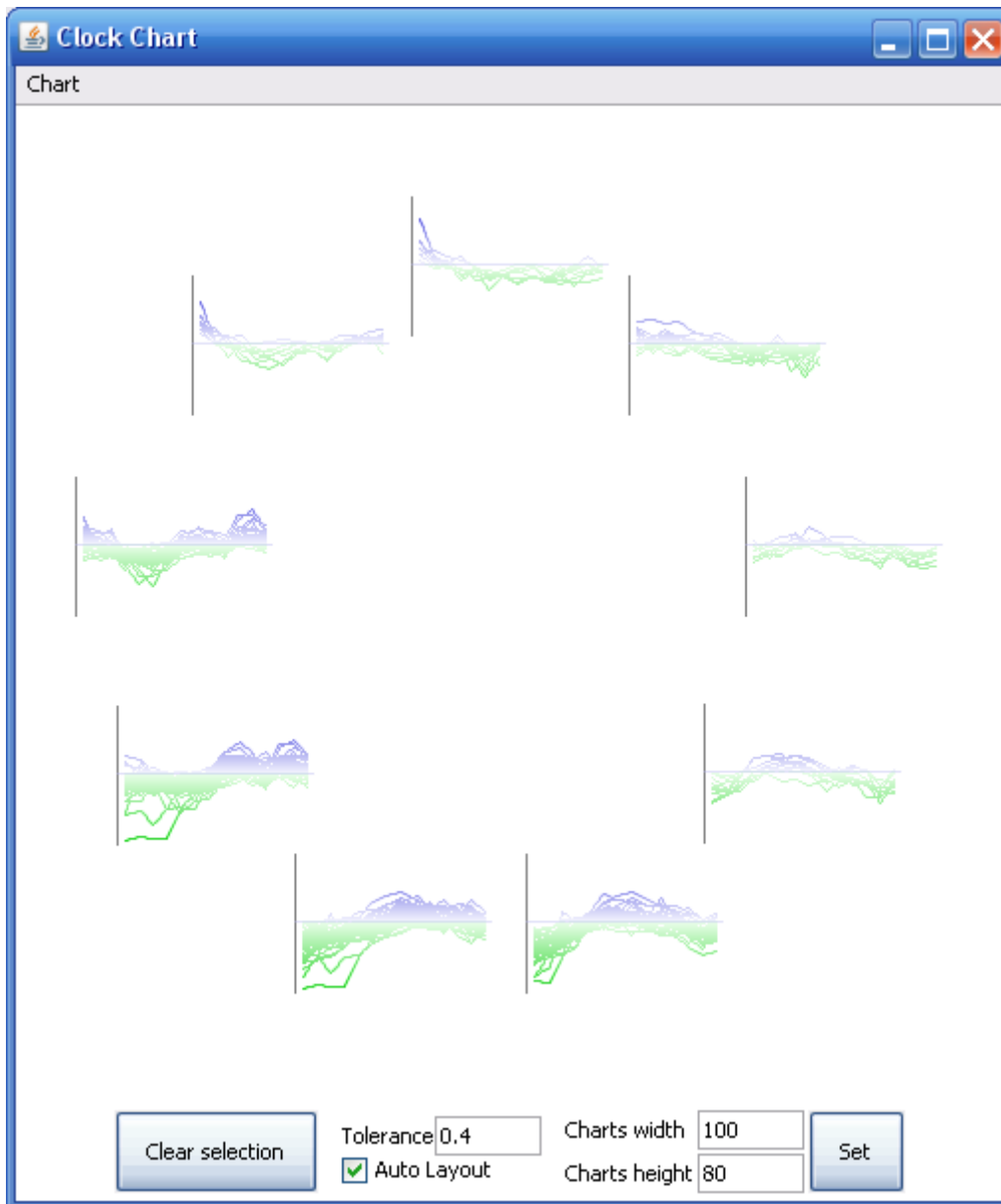
The clock chart

3.15 Array Plot
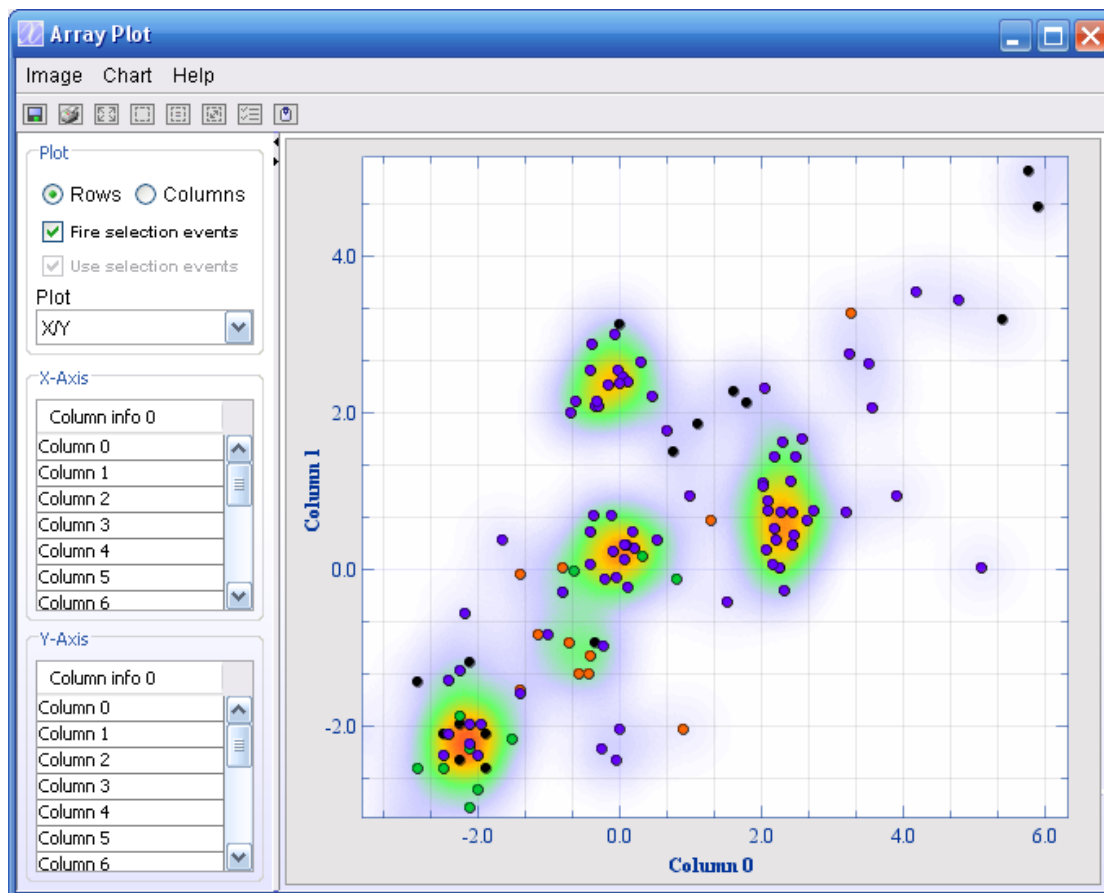
Array plot using density map.

The array plot allows you to create graphs of each profile in relation to another, or of each column (state) in relation to another. The array plot window has two areas on the left side used to select the profiles or states to be used as x- and y-axis. Use the Rows/ Columns selector above these to choose between plotting columns vs. columns, or rows vs. rows.

Save Image (🔲):

To save an image of the plot, press the **Save Image (🔲)** button or select the **Image | Save** from the Array Plot window menu bar. Select the location, name of the file and file format. Click Ok.

Print Image (🖼):

To print the plot press **Print Image (🖼)** button or select **Image | Print** button from the Array Plot window menu bar.

To zoom in on an area of interest press **Zoom In (🔳)** button**,** then click and drag out a selection box on the plot. To zoom back out again press **Zoom out (🔳)** button or select **Chart | Zoom out**.

Shadow Unselected (☑):

Shadow Unselected (☑) is only useful when the Rows radio button is selected. The selected profiles will be shown in full color, while the other profiles will fade to grey color.

Create Selection (▤):

Create Selection (▤) is only useful when the Rows radio button is selected. To create a selection press the **Create Selection (▤)** button. There are two different Frame Methods to use; ⬚ **Square** and ⬭ **Lasso.** When using the ⬚ Square method, click and drag the mouse around the area you wish to select. The ⬭ Lasso method lets you draw a line around the points you want to select. It is also possible to color the selection area with a color from the list in the Frame Method pull-down menu.

It is also possible to select genes from a Gene Graph (Section 2.2.7) window. Open a Gene Graph (▬) from the J-Express Pro tool bar, and select genes from the list. If the Shadow Unselected (☑) has been selected in the Array Plot, the genes selected in the Gene Graph will now be shown in full color in the Array Plot.

Copy Clip Image to Clipboard (📋):

To copy the image in any of the tabs to clipboard, click the 📋 button.

Fire Selection Event - check to update the chosen rows when selecting points in the graph.

Use Selection Event - check to update selected rows when they are selected from another component, such as Gene Graph.

Customizing the Array Plot

Select **Chart | Chart Settings** from the Array Plot menu bar. Another way to bring up the Chart properties window is by right clicking on the Array plot. For more information see Section 3.8.2.

All changes made in the Array properties window take effect as soon as you click **OK**. To set the current settings as default click the **Set Defaults** button.

3.16 Dataset Filtering
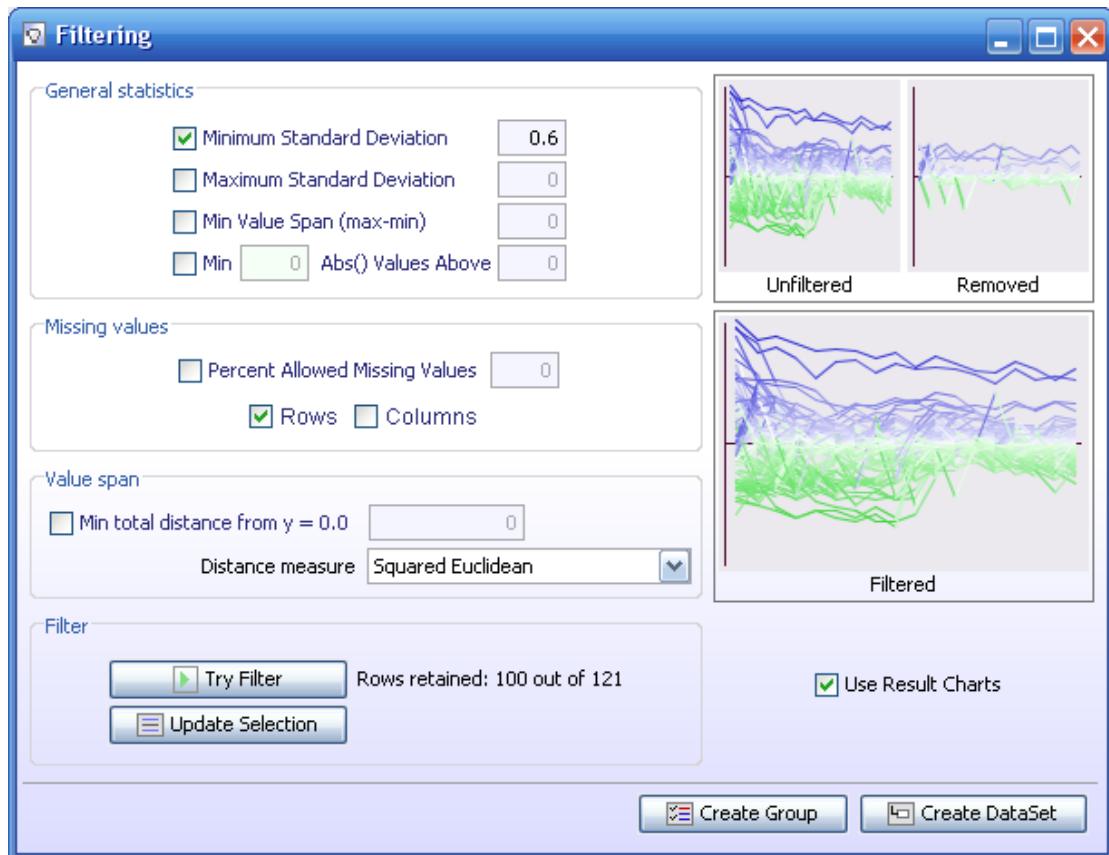
J-Express Pro has several methods for filtering a dataset. To access these, click the **Filter Data Set ([ ])** button on the J-Express Pro tool bar or select **Dataset|Filter Dataset** from the J-Express menu bar, with the node you wish to filter selected in the project tree. The Filter dataset window has several check boxes that allow you to activate or deactivate the various filter types. Note that you can use several filters at once.

Filtering options:

- **Minimum Standard Deviation** – check this box and enter a value to set the minimum allowed standard deviation that a profile can have to pass the filter.

- **Maximum Standard Deviation -** check this box and enter a value to set the maximum allowed standard deviation that a profile can have to pass the filter.

- **Min Value Span (max-min)** – check this box and enter the Minimum value span that a profile can have to pass this filter.

- **Percent allowed Missing Values** – check this box and enter a percentage value to only allow profiles that have less missing values (in percent of total points of the profile) than this percentage value.

- **Min total distance from Y=0.0** - check this box and enter a value that only allows profiles that have at least that great a distance from a profile that is 0.0 in all columns. Select the distance measure to use from the **Distance measure combo box.** Basically this allows you to filter profiles that are not differentially expressed. .

Click the **Try Filter** button to see how many profiles are filtered by the current settings. The number of rows retained (i.e. not excluded by the filter) is shown next to the button.

Click the ▤ **Update Selection** filter to select all profiles not excluded by the current filter settings. This selection takes effect in all windows in J-Express Pro, so if you for instance have a Gene Graph window open simultaneously, the selected profiles can be highlighted using the shadow unselected feature, etc.

Click the ▦ **Create Group** button to create a new group based on the profiles retained by the current filter settings.

Click the ▣ **Create Dataset** button to create a new node in the project tree containing the profiles retained by the current filter settings.

Click the **Close** button to close the **Filter** dataset window.

3.17 Creating a Sub data set
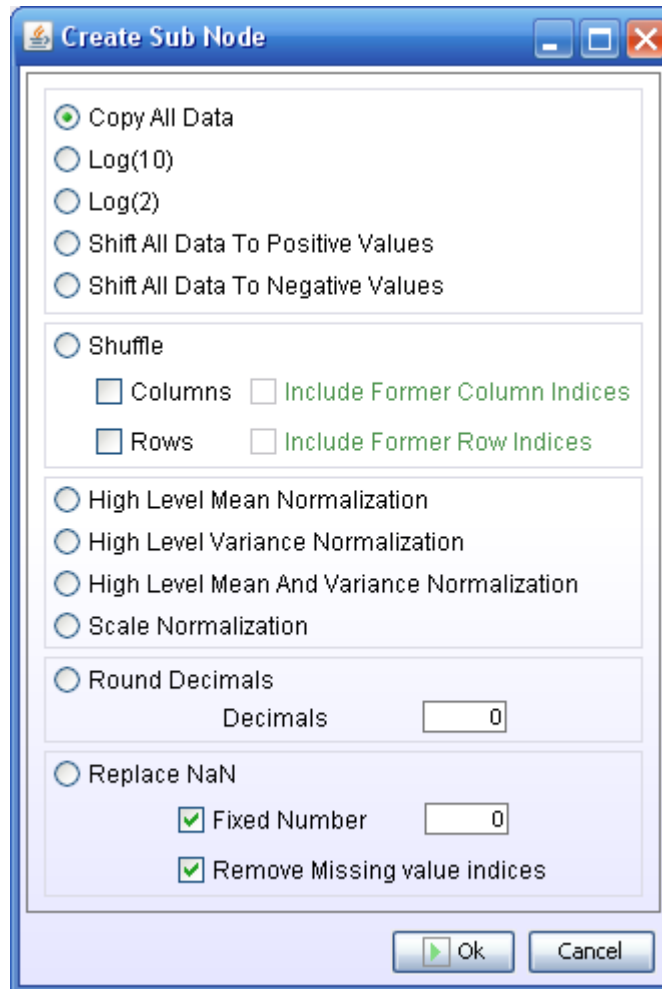
J-Express Pro provides a number of methods of modifying a dataset to suit your needs. To create a sub data set of a currently selected node in the project tree, click the **Create Sub Data Set** button (![icon]) from the J-Express Pro tool bar, or select **Dataset | Create Sub Data Set** from the J-Express Pro menu bar. The **Create Sub Node** window will open.

Select the operation you want to perform on the data by clicking the radio button next to it, and then click **Ok**. A new node containing the result of the chosen method will be created below the current node in the project tree.

- **Copy All Data** – this method simply copies all the data in the currently selected node.

- **Log(10) Transform All Data** – this method transforms the currently selected dataset into its logarithm (base 10). The data cannot contain any negative values.

- **Log(2) Transform All Data** – this method transforms the currently selected dataset into its logarithm (base 2). The data cannot contain any negative values.

- **Shift All Data To Positive Values** – this method shifts the entire dataset a constant amount along the Y-axis so that no profile contains a negative value.

- **Shift All Data To Negative Values** – this method shifts the entire dataset a constant amount along the Y-axis so that no profile contains a positive value.

- **Shuffle Columns/Rows** – this method shuffles columns or rows, respectively, based on a random algorithm. Check the **Include former Column/Row Indexes** to keep this information in the new node as a reference.

- **High Level Mean Normalization** – this method performs a high level mean normalization on the data in placed in the new node.

- **High Level Mean and Variance Normalization** – this method performs a high level mean and variance normalization on the data in placed in the new node.

- **Scale Normalization -** this method performs a scale normalization on the data placed in the new node.

  See

  Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation.
  Yang YH, Dudoit S, Luu P, Lin DM, Peng V, Ngai J, Speed TP.
  Department of Statistics, Helen Wills Neuroscience Institute, University of California, Berkeley, CA 94720-3860, USA.
  Nucleic Acids Res. 2002 Feb 15;30(4):e15.

- **Round Decimals -** this method creates a dataset with values rounded to the given number of decimals. Keep in mind that if this number is lower than the number of number of Maximum Fraction Digits (see Section 3.1.12), the rounded decimals will be replaced by zeros.
- **Replace NaN -** NaN values in your dataset can be replaced by a fixed number. Check Fixed Number checkbox and enter a number in the text field provided. If you want to keep the Missing Value Indices, uncheck the Remove Missing Value Indices checkbox.

3.18 Annotation manager
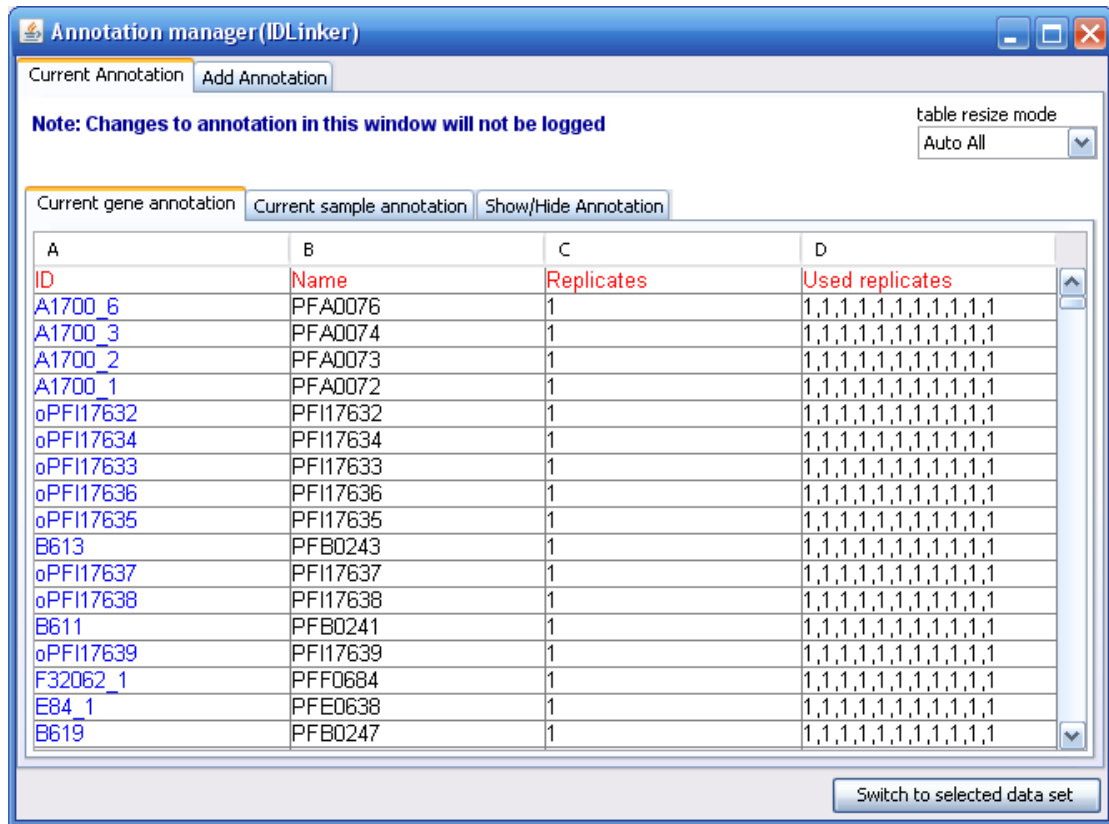
The annotation manager component can be used to modify, add or delete annotation on genes and samples. You may double-click any cell to change its value. Right-click the table to add annotation columns or delete existing columns.

The Current annotation tab shows the annotation currently in the seleted data set. The add annotation tab lets you map tabular annotation files to the selected dataset. You can paste any annotation from external applications such as excel into the "current annotation" table. This is the best way to add annotation if the order of your new annotation equals the order of the existing annotation. If the orders are not the same, you can map new annotation through a common key (see below).
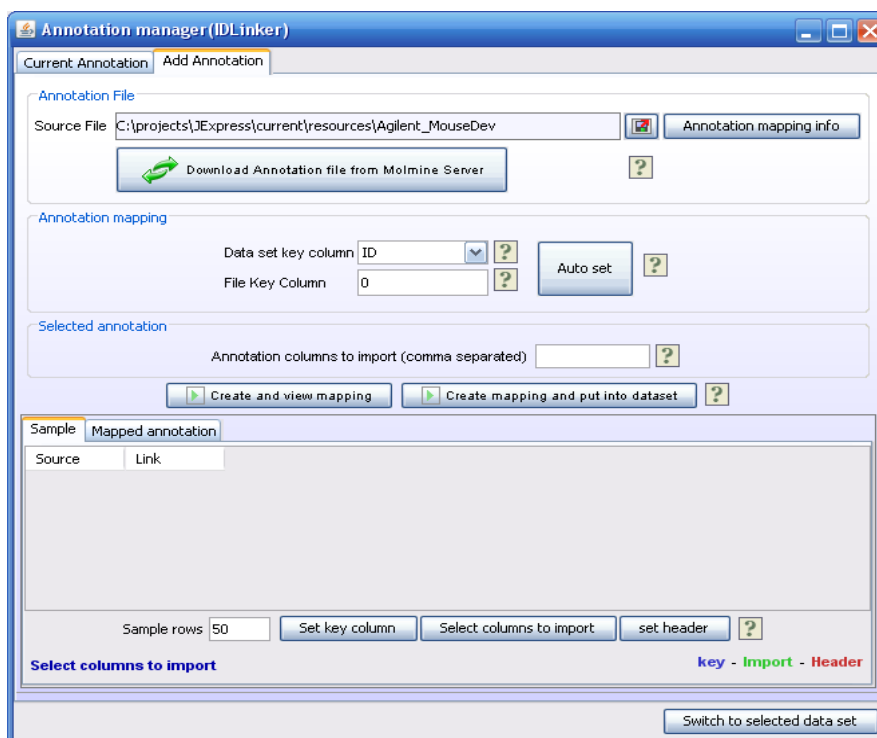
If you have a tab-delimited text file containing a key column that exists in your data set, you can use the ADD ANNOTATION component. The source file is the tab-delimited text file, and the **Dataset ID column** should contain the keys in your dataset that map to the keys in the file specified in the **File key Column** input field ("1" is the first column in this file). The **File Import Columns** are the column numbers of the columns containing the identifiers you want to import separated by a comma..

ANNOTATION MAPPING INFO reads the file specified and count occurrences of the annotation in the data set. This can help you locate common annotation keys in the file and your data set.

AUTOSET sets the dataset key column and the file key column by looking for common occurrences of annotation in the file and in the dataset.

When a file is specified and the *auto set* button is clicked, a number of rows (specified by the Sample rows input field) from the file are inserted into the sample table. The key column in this file is marked in blue. Whenever you click on a column in this table, the column will be added to the columns to import and marked green. By clicking on a selected green column, it will be deselected. J-Express will also look for an annotation header by counting occurrences of certain key words and mark this annotation header row in red. if you want to remove this header or specify a different row, click the **set header** button. You can continue selecting columns to import by clicking the **set columns to import** button. To specify a different column as the key column, click the **set key column button** and the new column.

There are two ways of putting the mapped annotation into your dataset:

Create and view mapping opens the mapped annotation table and previews the mapped annotation. you can then click the **put annotation in dataset** button to add the new annotation to the selected dataset.

Create mapping and put into dataset directly maps the new annotation and adds it to the dataset. The **current gene annotation** table is then opened for viewing.
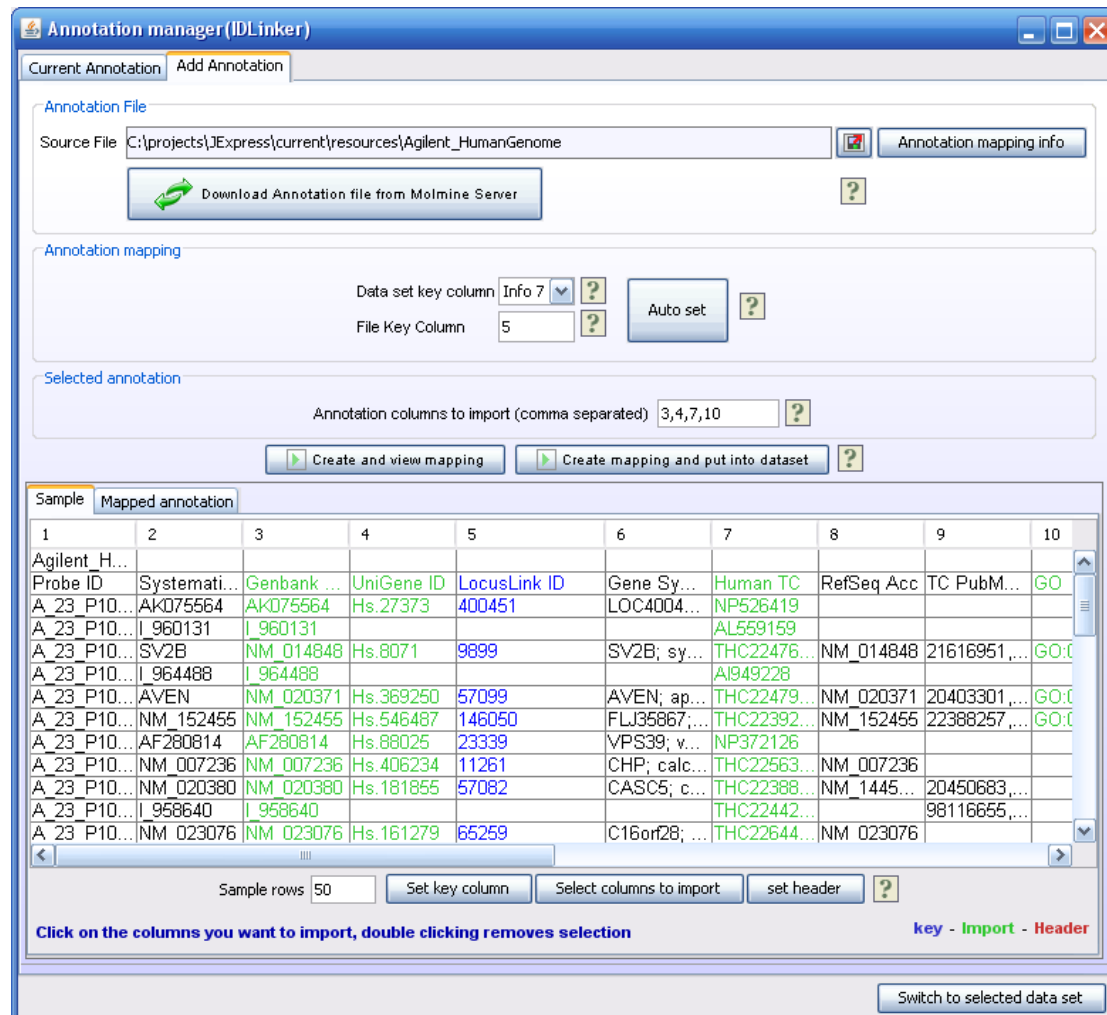
In the example above, we have loaded a file with Agilent annotation. The *Info7* column of annotation in our dataset contains the same annotation as the 5th column in the loaded file (both are locus link Ids). These two columns are chosen to link the annotation in the dataset with the annotation in the file. We click the *Select columns to import* and click on column 3,4,7 and 10 in the table because we want to import these into our dataset.

Clicking the *Create mapping and put into dataset* will map the selected annotation and put it into the dataset. Clicking the *create and view mapping* vill put the linked annotation into the *Mapped annotation* tab for viewing.

3.19 Search and Sort
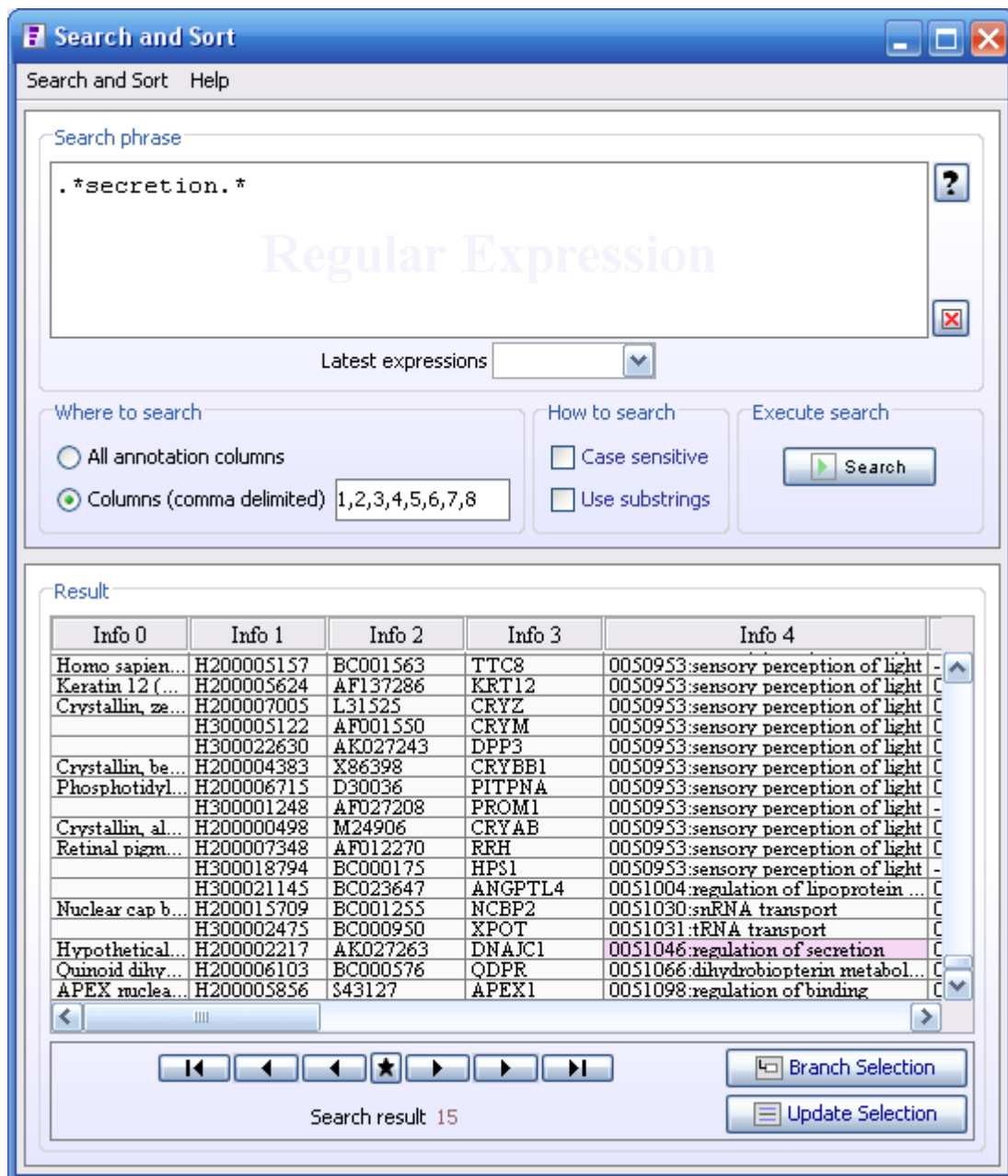
If you need to sort or locate profiles based on the identifiers in your dataset, use the Search and Sort window. Open this window by clicking the **Search and Sort** button (

) on the J-Express Pro tool bar, or selecting **Data Set** | **Search and Sort** from the J-Express menu bar. <u>Note:</u> This window is intended for use alongside other windows such as the Gene Graph and Group windows.

The **Search Phrase** text area lets you enter a query text. The query can be a simple text string, or a Regular Expression. For more information on regular expressions and a short syntax reference click the **?** button to the right of the text area. Check the **Case sensitive** box to make searching case sensitive (i.e. differentiate between

uppercase and lowercase letters). Previous searches can be accessed by using the pull down function of the **Latest Expressions** combo box.

If you want to search for your search phrase in all columns, select the **All Columns** radio button. If you only want to search particular columns, select the **Columns (comma delimited)** radio button, and type the column numbers, separated by commas, in the text field below. Press the Search button.

Hits from the search are highlighted in the spreadsheet below. Use the arrow buttons below the spreadsheet to move between the hits. To move to the first hit, click the I◀ button. To move to the previous hit and add it to the current selection, click the blue ◀ button. To move to the previous hit without adding it to the selection click the ◀ button. To add all hits to the current selection click the * button. To move to the next hit click the grey ▶ button. To move to the next hit and add it to the current selection, click the blue ▶ button. To move to the last hit in the search click the ▶I button.

If you click a column header on the spreadsheet, the column will be sorted. To invert the sort, click the column header again. Selections can also be made directly in the spreadsheet.

To branch off the selection and adding it to the project tree, press ⊟ **Branch Selection** button. This will add a new node under the

Click the **Update Selection** button to select all profiles matching the search phrase (This is the same as pressing the **star** button).

3.20 Chromosome view
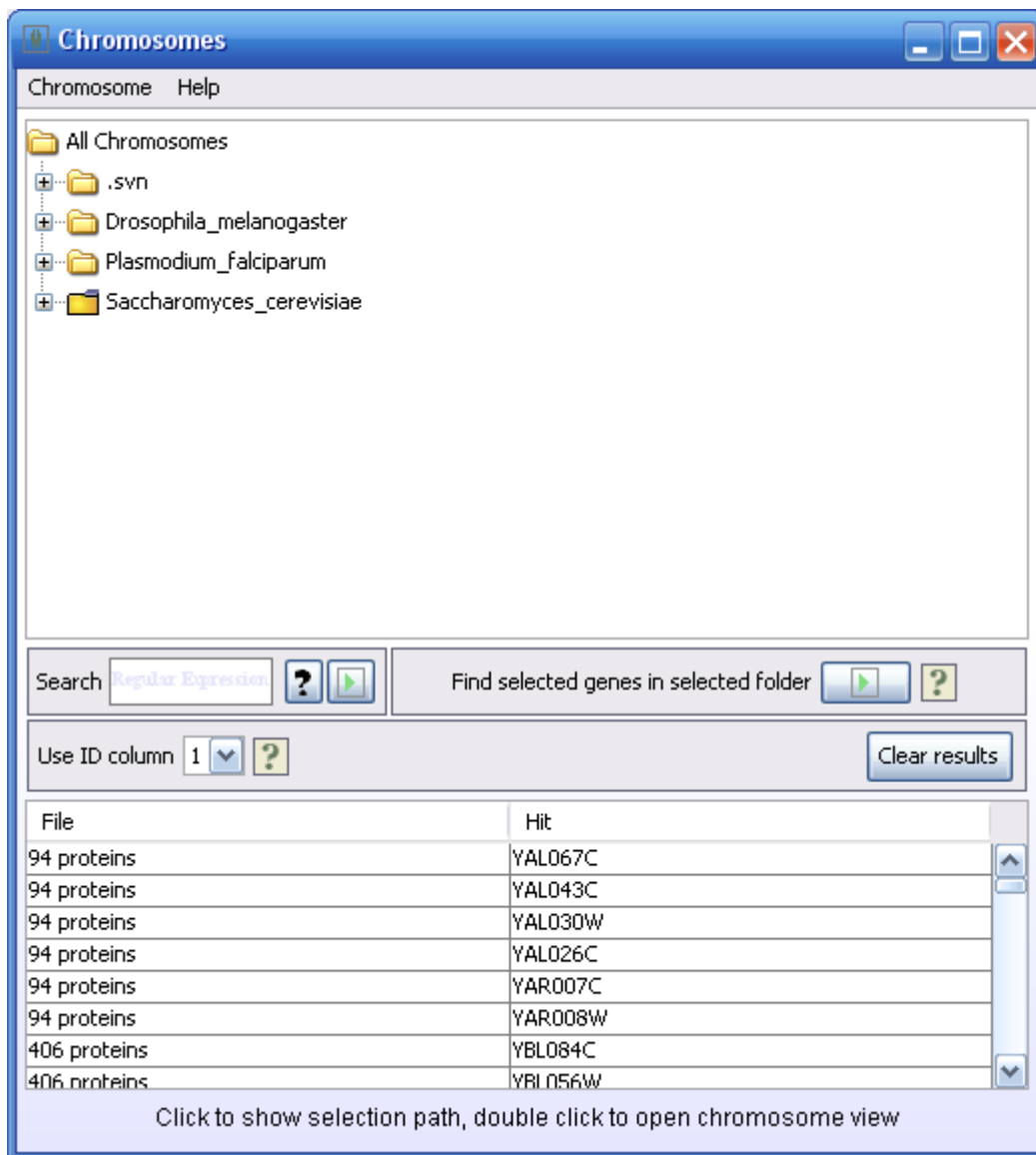
To open the chromosome view, select the node in the project tree that you want to analyze. To open this tool, select **Methods | Chromosome View** from the J-Express Pro menu bar, and a window with folders containing chromosomal data will open. There are several ways to move ahead to open the chromosome view.

Chromosome folders:

Moving through the chromosome folders and double clicking on any of the files with the file extension .ptt will open a **chromosome view** window showing the chromosome you clicked..

Search:

Select the folder you want to search. Right-click it and select **Set Selected Folder**. The selected folder will be marked yellow.

To search for one or several genes enter the name or synonym name explicitly or using a regular expression. Click on the **question mark(?)** for more information on regular expressions. Press **enter** or click on the **run(** ▶ **)** button. The result of the search will appear in the list below. Double click on any of the hits to open the **chromosome view**. Hits will be marked in the chromosome view.

Find Selected Genes in Selected Folder[▶]:

Select the folder you want to search. Right-click it and select **Set Selected Folder**. The selected folder is marked yellow.

If some genes have been selected in any of the other J-Express Pro components, eg. in the Gene Viewer, these genes can be located by clicking the **Find Selected Genes in Selected Folder[▶].** Make sure you select the correct **Use ID Column.** Set this column to the one in your dataset that contains the gene names.

The result of the search will appear in the list below. Double click on any of the hits to open the **chromosome view**. Hits will be marked in the chromosome view.
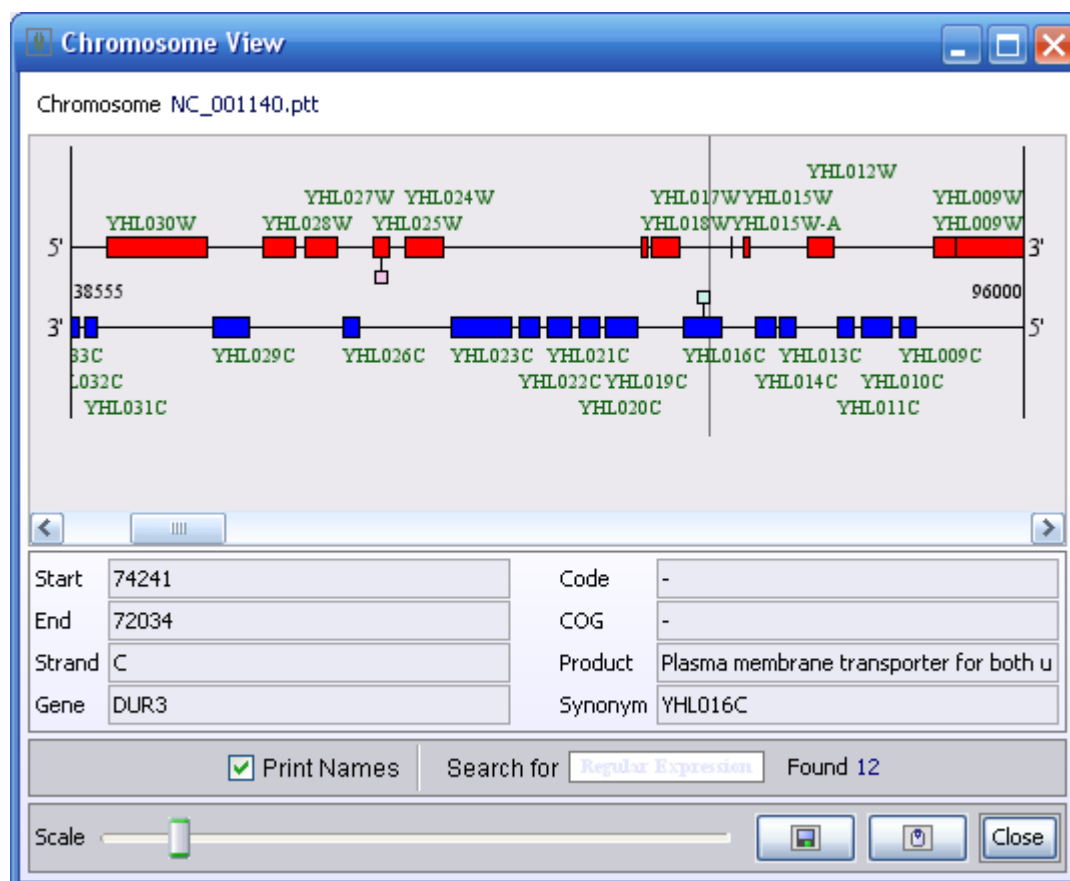
Adding Chromosomes

You can add chromosome files by downloading ptt (protein table) files from the genebank database and put them in your external folder (jexpress/resources/external). These files are located on the genebank ftp site and the various genebank mirror sites.

For instance, you want to add or update the chromosome files for D. Melanogaster; go to the ftp site:

ftp://ftp.ncbi.nih.gov/genomes/Drosophila_melanogaster

and select all the .ptt files. copy all these to the folder called \jexpress\resources\external\Drosophila_melanogaster or wathever name you may prefer for the folder. remember that the folder with the ptt files must be located somewhere under the external folder.

3.21 Correspondence Analysis
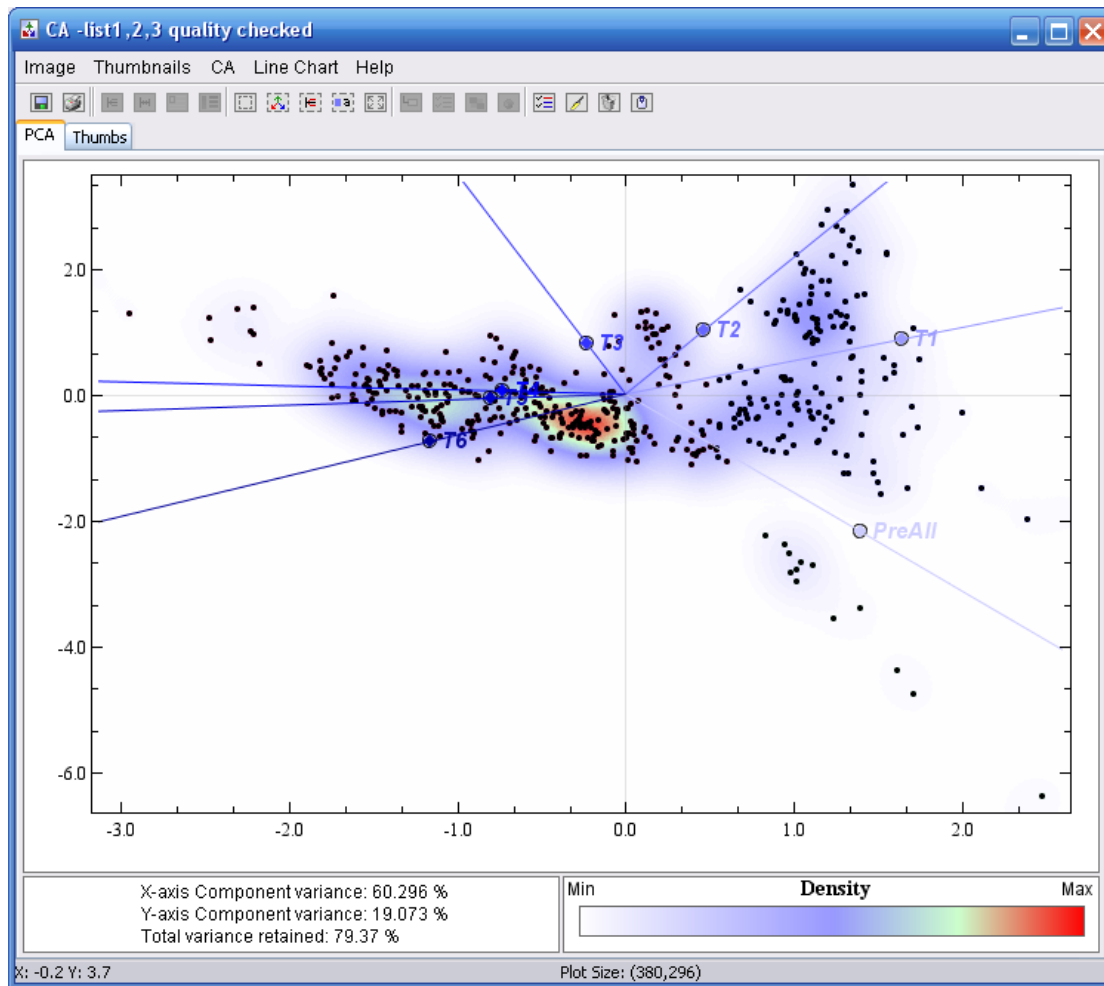
To perform correspondence analysis on a dataset click the **Correspondence Analysis** button ( ) from the J-Express Pro tool bar, or select **Methods|Correspondence Analysis** from the J-Express Pro menu bar. Note that Correspondence Analysis can only be performed on a dataset that exclusively contains positive values.

The result of the correspondence analysis is shown in a window that has common functionality with the PCA window. Refer to section 3.6 for a description on how to use this window. The only differences to PCA are that points are added and labeled for each column, and that the PCA-specific options are unavailable.

You can use the CA menu to specify the component to show in the plot.

Plot CA text settings – Set properties such as font and font transparency for the samples (The colors are set in the group controlle)

Toggle CA text – toggle the text for the samples

Toggle CA median marks – toggle on and off a line going from origo th the mean of the samples in each group. Create sample groups with the *create groups tool (Data Set menu and create groups)*

Please refer to the following paper for method explanation:

Correspondence analysis applied to microarray data.
Fellenberg K, Hauser NC, Brors B, Neutzner A, Hoheisel JD, Vingron M.
Proc Natl Acad Sci USA. 2001. 98(19): 10781-10786.

3.22 Feature Subset Selection and ANOVA
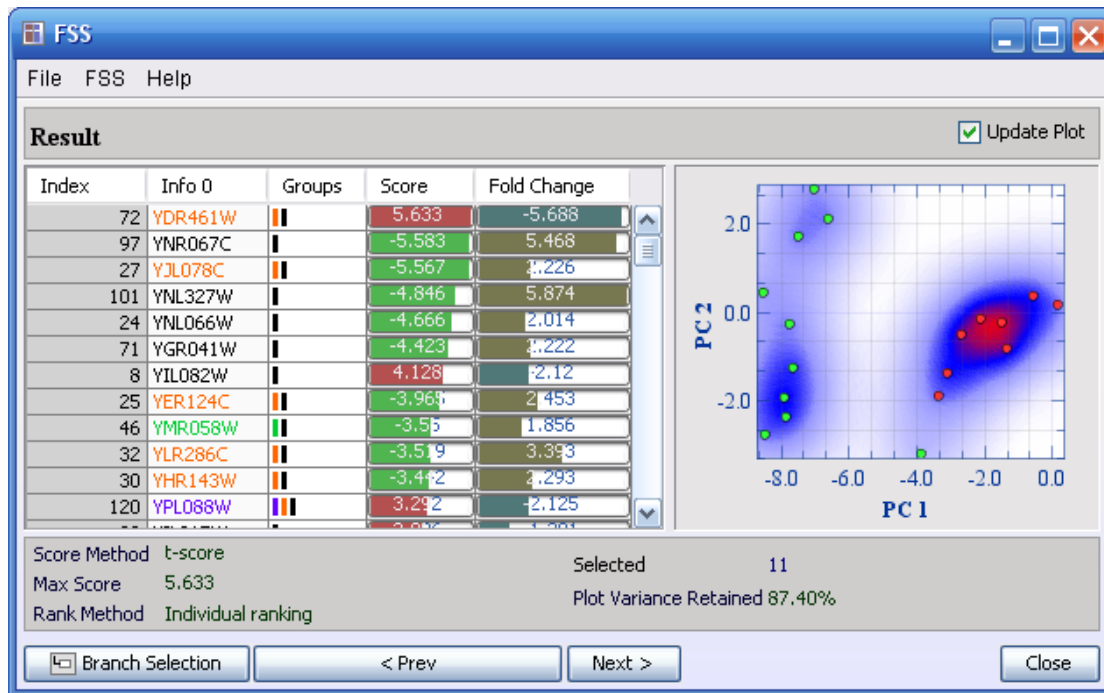
Feature subset selection will basically find the genes that best divide one group of genes from the rest, or divides multiple groups. To run this method, one or more column groups must be defined.

To perform Feature Subset Selection (FSS) on a dataset you need to make sure that there is at least one column group defined in the dataset. Then click the Feature Subset Selection (

) button from the J-Express Pro tool bar, or select Methods | Feature Subset Selection/ANOVA from the J-Express Pro menu bar. This will open a window that allows you to select one or two groups to perform the feature subset selection analysis on.

Check the box(es) in the Active column to select the group(s) for analysis.

The other columns provide additional information for the column groups, namely group name, color, and member count. Click the **Next >** button to set the parameters for the analysis. In the first window of the **FSS** you can select which method to use to discriminate between the classes (**FSS** and **ANOVA**). These methods produce very similar methods, but unlike **ANOVA**, **FSS** is based on statistical tests such as t-test which limit the classes to be separated to 2. For **ANOVA**, you can also select to display scores as p-values and out of these scores you can get a False Discovery Rate (**FDR** for your selection. The caclulation of the **FDR** is based on the Benjamini/Hochberg methods and is therefore based on a subset of the list. Choosing the top 10 p-values in the list will automatically update the **FDR** for a list with the 10 top scoring p-values. The plot generated from selection will be the same as for **FSS** (See below).

For FSS, you can set the parameters in the second window. Select the Score and Rank algorithms to be used by selecting the desired algorithm in the pull down menus. You can choose how many of the highest scoring profiles by selecting a value in the Result pull down menu. Click the **< Prev** button to go back to the group selection window, or click the **Next >** button to complete the **FSS** analysis.

The Result window has two main areas. On the left the highest scoring profiles are shown. The number of profiles in the list is based on the value set in the Result pull down menu in the Parameters window. Additional defined profile information is also shown, as well as group membership colors and the profile index. The **FSS** score of a profile is shown as a colored bar, where a longer bar indicates a higher score. Multiple profiles can be selected in the list, and these profiles will then be displayed in the plot on the right side of the window.

The plot will show a gene/gene plot if one row is selected in the table. Gene1/gene2 if two rows are selected and a principal component projection if more than two rows are selected. To see the profile of the selected genes, open a line chart component and click on **Shadow Unselected (	)** button.

You can customize the appearance of the plot by right clicking on it.

Fill lets you choose the background color of the FSS plot. The options are:

- One color – click the Background color box to select a new background color for the FSS plot.

- Density map – uses a spectrum of colors to show the density of points in an area.

    Density Map options:

    These options become available when the density map is selected as the fill type.

    Density Map Colors – allows you to change the color of the highlights. To change a color in the FSS color range simply click one of the small boxes over the spectrum. This brings up a color selection dialog where you can choose the color you want. Click OK, and the color range will change to accommodate your changes.

    Density area – This allows you to set the size of the area a single dot influences on the density map. To make the influence of a dot less, move the slider to the left, to increase the influence of a dot move the slider to the right.

    Number of Colors – this option sets the number of colors to be used to generate the density map. A small number of colors limits, and in some cases removes, the density map for dots lying in areas of low density. In addition the transition between colors becomes less gradual. Move the slider to set the desired amount of colors to be used.

Paint Threshold – This option sets a threshold value for the amount of dots in an area. If this threshold is exceeded the dots in that area are removed. This frequently helps show the structure of the Density Map. Move the slider to set the desired threshold.

- Gradient – Two colors are combined to create a smooth color gradient. Click the two colored boxes to choose the desired colors. Use the Gradient Type menu to select the type of gradient. Diagonal forms a color gradient from the upper left to the lower right corner; Top-Bottom forms a color gradient from the top of the plot to the bottom.

- External Picture – Use the file selection dialog to select the image file you wish to use as a background for the plot. Selecting Stretch will stretch the image to fit the plot. Selecting Tile will repeat the image in a tile pattern if it is too small to cover the entire plot.

- Tiles – Six additional patterns you can use for your plots.

Spot size lets you set the size in pixels of the FSS points.

Circular Spots – check this box to use circular FSS points.

Framed – Checking this box adds a frame around each dot.

Title – enter a title for your chart in this box, if needed. It will appear at the top of the chart.

Axis Value Span lets you set the maximum and minimum values for each axis. Uncheck the Force Endlabels box to turn off the automatic endlabels generated by J-Express Pro. Click the Reset button to center the chart on origo.

Chart & Axis color – click these colored boxes to set the background color for the area outside the main chart, and the colors used for the axis.

X- and Y-axis options

- Title allows you to name each axis. The name will appear on the left side of the chart for the y-axis, and on the bottom of the chart for the x-axis.

- Minor tics set the amount of minor tics between each major tic on the respective axis.

- Tics on both ends – check this box to have tics on the opposite edge of the plot from the axis, in addition to the tics on the axis.

Grid lets you set options for the plot grid.

- **Paint Grid** – check this box to toggle display of the grid on. Uncheck it to toggle display of the grid off.

- **Grid Color** – select the desired color for the grid by clicking on this box and choosing a color from the dialog that appears.

- **Grid Transparency** – Use this slider to set the transparency of the grid, relative to the background.

- **Null color** sets the color to be used to indicate that the value represented is a replaced erroneous value.

Click the **Prev >** button to return to the Parameters window. Click **Close** to close the FSS window.

3.22.1 Score methods

J-Express pro now include the following methods to test for differential expression between two microarray experiment states:

t-score

The t-score is the two sample t-statistic. Given means of two experiment classes m_1 and m_2, the pooled standard deviation estimate s_p, and the number of experiments in each class n_1 and n_2, the score is computed by the formula

$$\frac{m_1 - m_2}{s_p \sqrt{\dfrac{1}{n_1} + \dfrac{1}{n_2}}}$$

Golub score

The Golub score is named after the widely referenced paper by Golub et al. [4]. This scoring method is often referred to as the "signal-to-noise" ratio. Given means of two experiment conditions m_1 and m_2 and the corresponding standard deviations s_1 and s_2, the score value is computed by the formula

$$\left| \frac{m_1 - m_2}{s_1 + s_2} \right|$$

Between/within variance ratio

The between-to-within variance ratio reflects differences in class means relative to the variances in the classes. This score method was introduced by Dudoit et al. [3]. Given the class means m_1 and m_2, the grand mean $\overline{m}$ and the within class sum of squares ss_1 and ss_2 the score is computed by the formula

$$\frac{\left(m_1 - \overline{m}\right)^2 + \left(m_2 - \overline{m}\right)^2}{ss_1 + ss_2}$$

Wilcoxon z-approximation

The Wilcoxon z-approximation is a nonparametric score based on the Wilcoxon rank sum statistic. Given a decent number of experiments, the score is approximately standard normal distributed. The expression values are ranked and the rank sum W_a of the smaller

sample size is computed. Given the number of experiments in the smaller class n_a and in the larger class n_b ($n_a \le n_b$), the score is computed by the formula.

$$\frac{W_a - n_a(n_a + n_b + 1)/2}{\sqrt{n_a n_b(n_a + n_b + 1)/12}}$$

For further details, see for instance Bhattacharyya and Johnson [1].

3.22.2 Ranking methods

Given the score methods described above, genes can be ranked based on score if experiment classes are defined. J-Express include several variants for finding good marker genes, either by ranking gene by gene or by looking at combinations (pairs) of genes.

Individual ranking

This ranking method computes a score for each gene profile, and ranks the list of genes by score. The genes with highest (absolute) score are reported on top of the list.

Greedy pairs

The greedy pairs ranking method first ranks all genes by individual ranking. Subsequently the highest scoring gene g_i is paired with the gene g_j that gives the highest gene pair score. The gene pair score is computed by projecting the expression values of the two genes onto the diagonal linear discriminant axis, and then taking the score of the transformed data points. After the first pair has been selected, the highest ranked gene remaining g_s is paired with the gene g_r that maximizes the pair score, and so on. See Bø et al. [2] for further details.

All pairs

Unlike greedy pairs, this method examines all possible gene pairs by computing the pair score for all pairs. The pairs are then ranked by pair score, and the gene ranking list is compiled by selecting non-overlapping pairs, and selecting highest scoring pairs first. This method is computationally intensive, and may take a while to terminate. See Bø et al. [2] for further details.

[1] Bhattacharyya GK and Johnson RA: Statistical concepts and methods. Wiley, 1977.
[2] Bø TH and Jonassen I: New feature subset selection procedures for classification of expression profiles. Genome Biology, 3(4):research0017.1-0017.11, 2002. Available online: http://genomebiology.com/2002/3/4/research/0017.1.
[3] Dudoit S, Fridlyand J, Speed T: Comparison of discrimination methods for the classification of tumors using gene expression data. Technical report no. 576, Department of Statistics, University of California, 2000.
[4] Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeeck M, Mesirov JP, Coller H, Loh ML, Downing JR, Caligiuri MA, et al.: Molecular classi_cation of cancer: Class discovery and class prediction by gene expression monitoring. Science 1999, 286:531-537.

Please refer to the following paper for method description:

New feature subset selection procedures for classification of expression profiles
Trond Hellem Bø , Inge Jonassen

Department of Informatics, University of Bergen, N-5020 Bergen, Norway

3.23 Significance Analysis of Microarrays (SAM)

Called	Info 1	d[i]	de[i]	Delta[i]	FSN[i]	FDR[i]	Fold Change	q-val[i]
1	1182	11.42	2.552	8.868	0.0	0.0	7.575	0.0
2	979	11.218	2.36	8.858	0.0	0.0	7.796	0.0
3	1652	10.322	2.262	8.061	0.0	0.0	3.946	0.0
4	2481	9.232	2.194	7.038	0.0	0.0	4.059	0.0
5	956	8.906	2.145	6.761	0.0	0.0	2.523	0.0
6	436	8.67	2.101	6.569	0.0	0.0	4.107	0.0
7	874	-8.514	-2.514	6	0.0	0.0	-5.288	0.0
8	3441	-8.199	-2.33	5.869	0.0	0.0	-2.491	0.0
9	3038	7.747	2.057	5.69	0.0	0.0	2.975	0.0
10	918	7.287	2.022	5.265	0.0	0.0	5.553	0.0
11	456	-7.264	-2.224	5.04	0.0	0.0	-2.061	0.0
12	3216	7.261	1.994	5.267	0.0	0.0	3.006	0.0
13	1219	-7.174	-2.153	5.021	0.0	0.0	-2.523	0.0
14	626	-7.104	-2.12	4.984	0.0	0.0	-2.397	0.0
15	1356	-6.876	-2.073	4.804	0.0	0.0	-3.885	0.0
16	1249	6.821	1.965	4.857	0.0	0.0	3.806	0.0
17	1099	6.79	1.937	4.853	0.0	0.0	1.967	0.0
18	907	6.728	1.917	4.811	0.0	0.0	2.359	0.0
19	435	-6.701	-2.042	4.659	0.0	0.0	-4.904	0.0
20	2220	6.654	1.898	4.756	0.0	0.0	2.896	0.0
21	2145	6.584	1.878	4.706	0.0	0.0	2.845	0.0

S0 0.04839
Pi0 0.47102

Selected 101
FDR for delta (2.904) = 0 %

Method Two Class Unpaired
Permutations 40

Branch Selection   < Prev   Next >   Close

SAM is a method that can be used to identify genes that are significantly differentially expressed. Each gene is assigned a score on the basis of its change in gene expression relative to the standard deviation of repeated measurements. The genes that have a score higher than some adjustable threshold are used to estimate the significance of the result. This is done by permuting the measurements to see how many genes comes up with a score above the threshold. The percentage of genes identified by chance is called False Discovery Rate (FDR). For more details on SAM, see

Significance analysis of microarrays applied to the ionizing radiation response. Tusher *et al* 2001

To perform SAM analysis, you need to define groups within your dataset.See the *create groups* for information about how to create groups.

The SAM analysis can be started from the **Methods** menu or by clicking the (**Significance Analysis of Microarrays**) button. In the window that opens, select the two groups to be compared by checking the boxes in the **Selection** column and click the **Next** button.

In the next window you can set the maximum number of permutations to be performed in order to calculate the FDR. In addition you need to tell J-Express whether your data values are **Linear** (non-logged), **Log2** or some other transformed values. Click **Next.**

In the **Result** window, you are presented with a table containing the genes of your dataset sorted according to their **score** in an ascending order. The score used by SAM is called d-score. The **Fold Change** for each gene is also presented.

The **Delta** value is the adjustable threshold used to select the differentially expressed genes. The genes that have a score higher than the delta value are used to calculate the FDR.

3.23.1 The SAM Plot

In the SAM Plot the observed relative difference **D(i)** is plotted against the expected relative difference **DE(i)**. The black line indicates the line for D(i) = DE(i). The two grey lines on either side of the black line, are drawn at **delta** distance from the black line. The grey lines show the selected threshold. Spots further away from the black line than the grey lines will be **Selected** as differentially expressed and used to calculate FDR.

Right-click in the plot area to see different plot options.

3.23.2 Plot options

Right-clicking in the plot area opens a menu where you can **zoom**, **save** and **print** the plot. **Properties** lets you, amongst other things, change the **title** of the **axis** and **plot**, and select different **font** and **colours**.

3.23.3 Outputting results

There are several ways of outputting the result from SAM. You can save the table containing the entire dataset with the score value and fold change value to a text file, save and print the plot, or branch selection to continue working with the selected genes in J-Express.

Click **File | Save Table** to save the table containing the list of genes with their scores and fold change.

There are two different ways of saving or printing the plot. You can either click **Save Chart** or **Print Chart** from the **File** menu, or you can **right-click** in the **plot area** to select the same options.

To continue working with the **Selected** genes, click the ⬓**Branch Selection** button. The new branch will be added to the J-Express project tree under the dataset you are working on. It will look like ✎SAM.

3.24 Gene Set Enrichment analysis

Gene Set Enrichment Analysis or GSEA is a supervised analysis method used to find statistically significant differences between sample groups defined by a priori defined gene groups. The gene groups can be loaded from a file with group definition or other analysis methods such as gene ontology.

See also the paper:
http://www.broad.mit.edu/gsea/doc/subramanian_tamayo_gsea_pnas.pdf

for detailed description of the GSEA method.

3.24.1 Background

Gene set analysis is used to look for sets of related genes that follow the same trends in the dataset. It can be performed on either categorical or continuous data. Categorical data is of the type "before" vs "after infection", while continuous data can be time series data.

Categorical
The traditional way of analysing data for differentially expressed genes has been to use statistics that look at each gene by itself. There are several advantages to analyse sets of genes rather than individual genes.

If the genes only change moderately it may be difficult to find significant change by looking at each gene separately. If, on the other hand, many genes belonging to the same gene set, e.g. immunity and defence, are changed, even moderately, this could be an interesting finding, and the a priori defined relationship between these genes gives more statistical power to detect such smaller changes (affecting a whole set of related genes) compared to a per gene statistic.

It has been common to do simple over representation analysis of for instance GO terms among genes found differentially expressed compared to the non-differentially expressed genes. One would then calculate a per gene statistic, rank the genes and select a cut-off on a certain number of genes or a certain p-value to divide the genes into differentially expressed and non-differentially expressed genes. The per gene individual statistic, and thus the gene expression values themselves, is only used to rank the genes in this approach, not to evaluate the gene sets themselves.

In contrast the gene set enrichment method does not depend on a cut-off, and use the gene expression values of the genes in the evaluation of a gene set. After ranking the genes according to some per gene statistic, the entire ranked list is used to assess how the genes of a gene set distribute across the ranked list. The score (statistic) of individual genes are taken into account when evaluating a set of genes for differential expression.

Imposing a hard cut off on a list of genes with smoothly decreasing statistical scores is bound to be an arbitrary choice, and introduces an artificial border that is

oversimplifying the biology. Genes in the area below the cut-off is easily missed that could exhibit the same behaviour as related genes in the list above the cut-off.

Continuous
Normally we cluster continuous data to search for genes that have similar expression profiles, and then we go through the genes belonging to a cluster to see if they share some common characteristics. The problem with this approach is that the decision on which cluster a gene is a member of may to some extent be arbitrary, depending on the clustering method, the number of predefined clusters and the initialisation of the clusters. Some genes that belong to the same gene set may therefore sometimes end up in the same cluster, while other times they end up in different clusters.

Another way continuous data have been analysed has been to search for genes in the dataset with a certain degree of similarity to a particular search profile. Obviously this creates a similar problem as the one described for categorical data; where do we set the cut off? How similar do a profile has to be to make it on to our gene list? All sorts of profiles exist in a data set and it is most likely going to be very difficult to set a clear cut threshold to say that a particular set of genes are similar to the selected profile, while the others are not similar. The resulting limit is therefore always going to be random.

By using a gene search profile and predefined gene sets it is possible to avoid the problems of clustering and setting a cut-off for similarity to a gene profile. We can also get a significance score for each gene set. All the genes in the dataset will then be ranked according to correlation with the search profile. Once the genes have been ranked, the gene sets are scored exactly like they are for categorical data.

3.24.2 Running GSEA

Select the dataset in the project tree that you want to run GSEA on

Select *GSEA* from the *Methods* menu, or click on the (*Gene Set Enrichment Analysis*) button
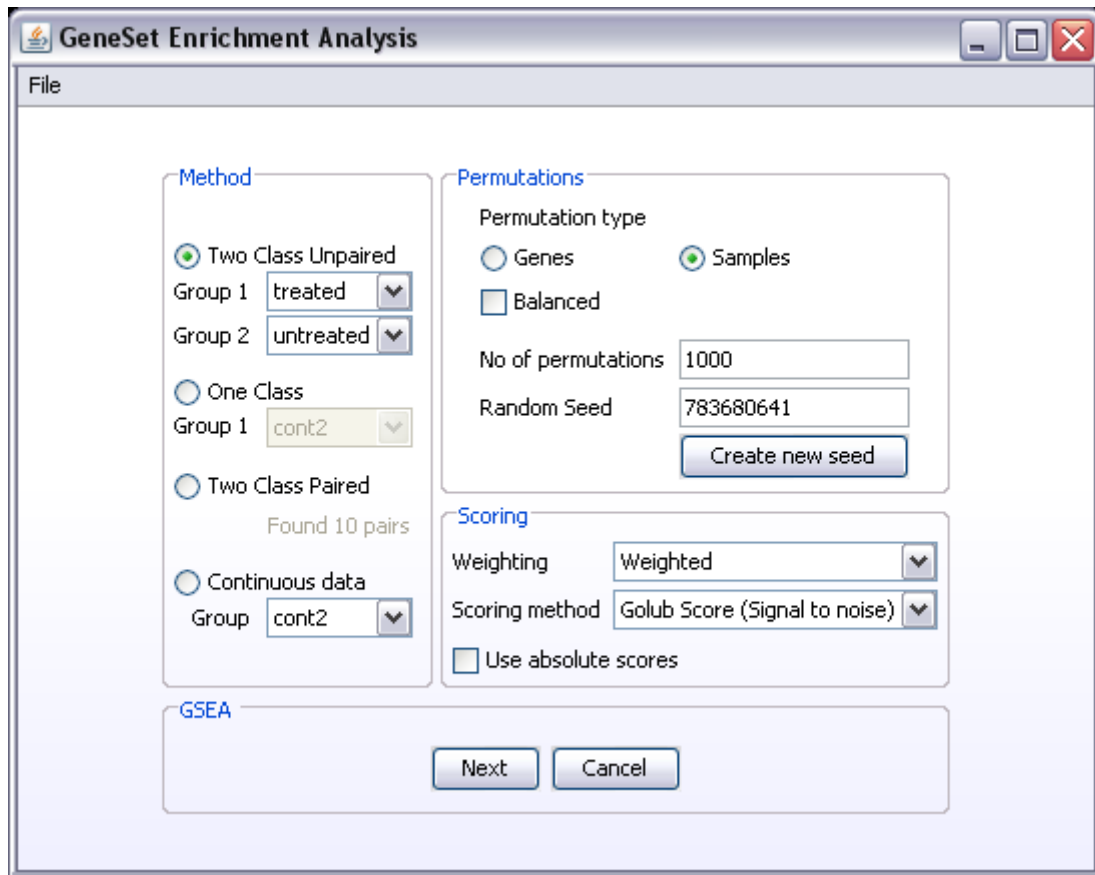
Select the appropriate analysis method:

1. **Two Class Unpaired**: select the two sample groups to be compared from the two drop down lists. If the drop down lists only contain one item (All) you have to define the sample groups from the *Create Groups* component.

2. **One Class**: select the sample group to be analysed from the drop down list. If all the samples in your dataset belong to the one sample group you want to analyse, you can select the item All from the dropdown list. If you want to test a sub-group of samples, and the drop down list only contain one item (All), you have to define the sample group from the *Create Groups* component.

3. **Two Class Paired** The number of defined pairs in the dataset will be listed under this selection. If no sample pairs are found, you have to define the pairs using the *Create Groups* component.

4. **Continuous data (e.g. time series):** The continuous method ranks the data according to its correlation to a profile that you define, or to a specific gene profile that you select from your dataset. Click on the *Create / Select Profile* button to set the search profile.

Next select the type of permutation you want to do. If there is enough samples in the dataset, sample permutations will give a more correct estimate of the background distribution of the data. If you have 5 or less samples in each group you are trying to compare a warning will appear saying you should consider gene permutations.

The number of permutations depend on type of permutation you are doing. Gene permutations will require more permutations that sample permutations. One way of assessing whether you have done enough permutations is to do the analysis again and compare the result. If they are not consistent, more permutations may be needed.

Next select a scoring method.

Weighting means that you value the scores at the top of the ranked list higher than the ones further down the list. An in-depth explanation can be found at http://www.broad.mit.edu/gsea/doc/GSEAUserGuideFrame.html.

The use of absolute scores means that the up and down regulated genes will be at the top of the list and the non-changing genes will be at the bottom of the list. If you do not use absolute scores the up-regulated genes will be in one end of the list, the down-regulated genes at the other end of the list and the non-changing genes will populate the middle of the list.

Click the **Next** button
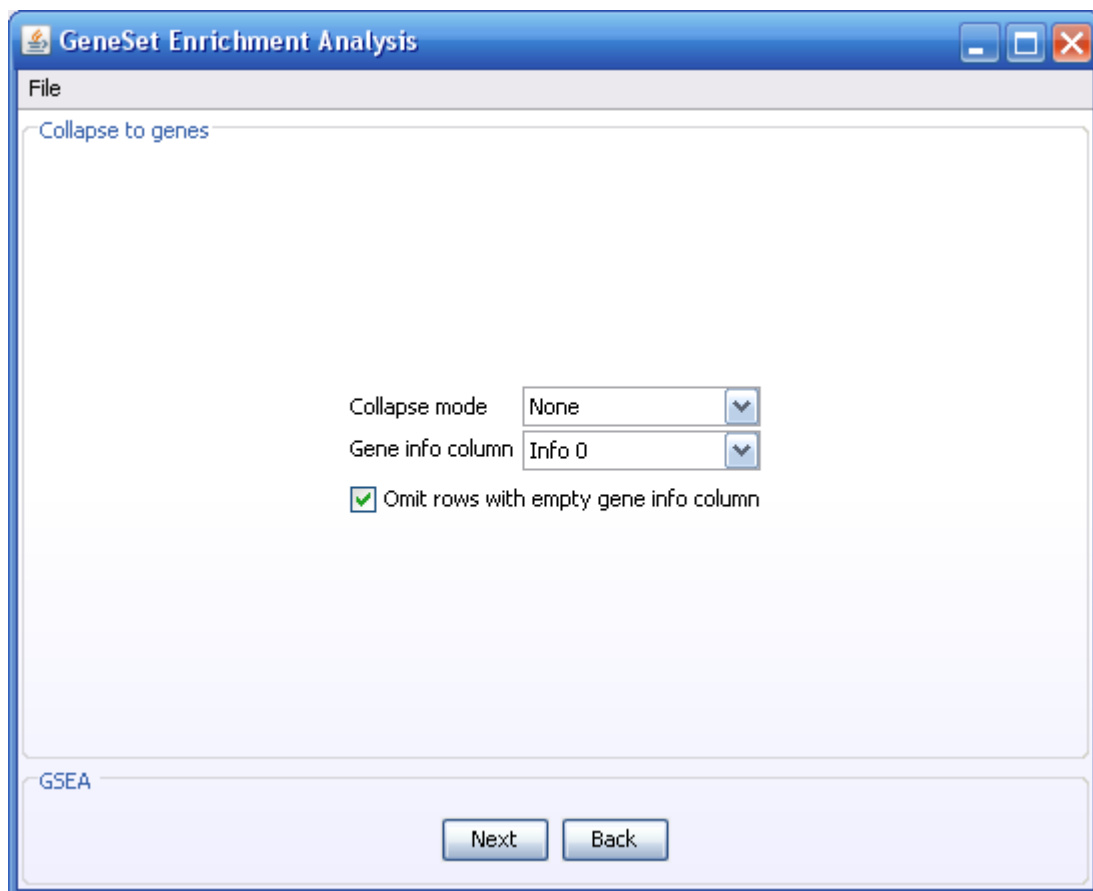
Some genes may only have one probe on the array while other genes may have multiple probes on the array. To avoid giving some genesets higher scores based on the number of probes its genes has, it may be a good idea find one profile that can represent the genes with multiple probes. This is called collapsing probes to genes. Creating the new profile can be done in different ways. Choose a collapse mode to select how to create the new gene profile.

- Collapse modes:

  - Max probe: of all the probes that map to the same gene the value of the probe with the highest intensity is selected

  - Median profile: the median value of all of the probes that map to the same gene is selected.

Select the column in your dataset that contains the gene id's.

Depending on which gene id you use for Gene info column there may be some blank entries. For instance if you use Gene symbol, not all hypothetical genes that contains probes on your array have gene symbols. These rows can be omitted. The chance that these genes have been mapped to a gene set is lower than for other known transcripts.

Click **Next**
NOTE: a new node called Collapsed to genes will now be added to the project tree and the new node is also automatically selected.

There are currently two ways of mapping your dataset to different gene sets. One is by using the GO tree and the other way is by importing predefined gene sets saved in a .gmt file

Gene Ontologies can be used as a basis for creating gene sets. To use *GO* to create gene sets select *GO Tree* as the *Gene Set Source* and click on the *map dataset to a GO tree* button.
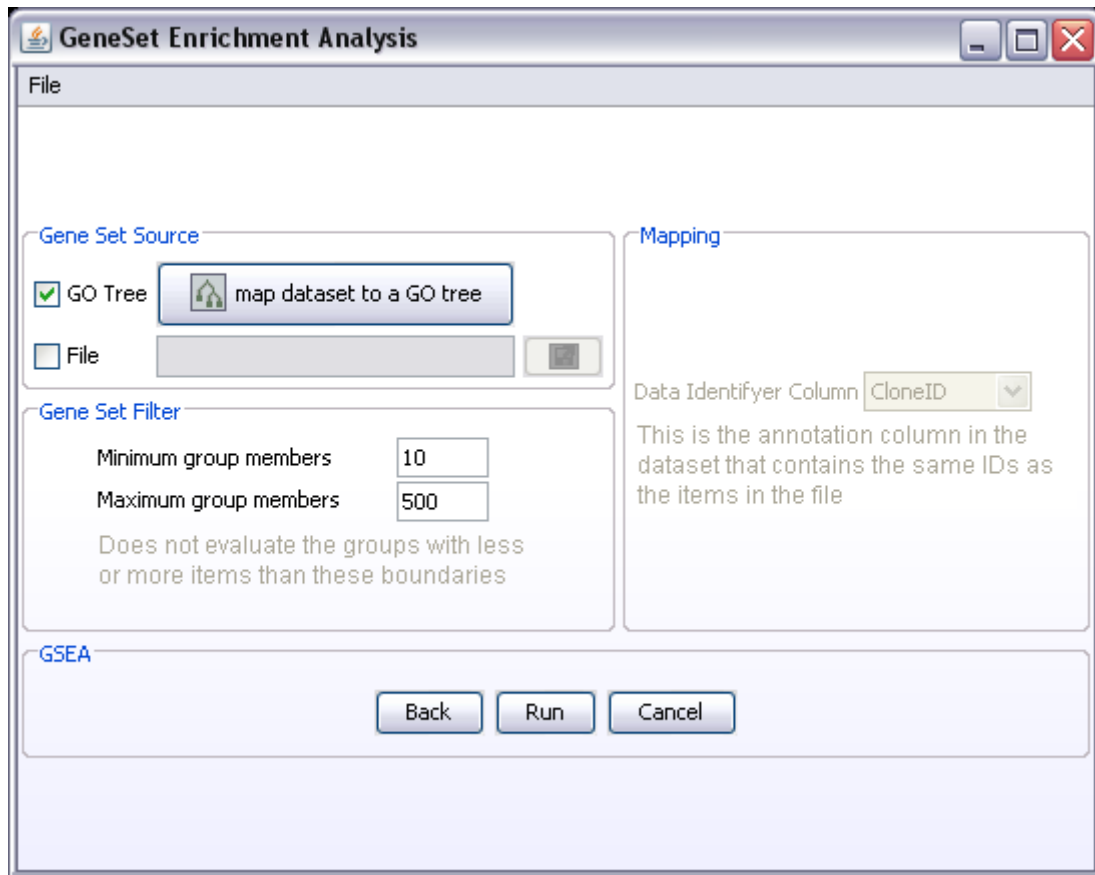
Gene sets saved to a .gmt file. The .gmt file format is explained at the end of this section. To import a .gmt file select *File* and click the 🖼 button to locate the file.
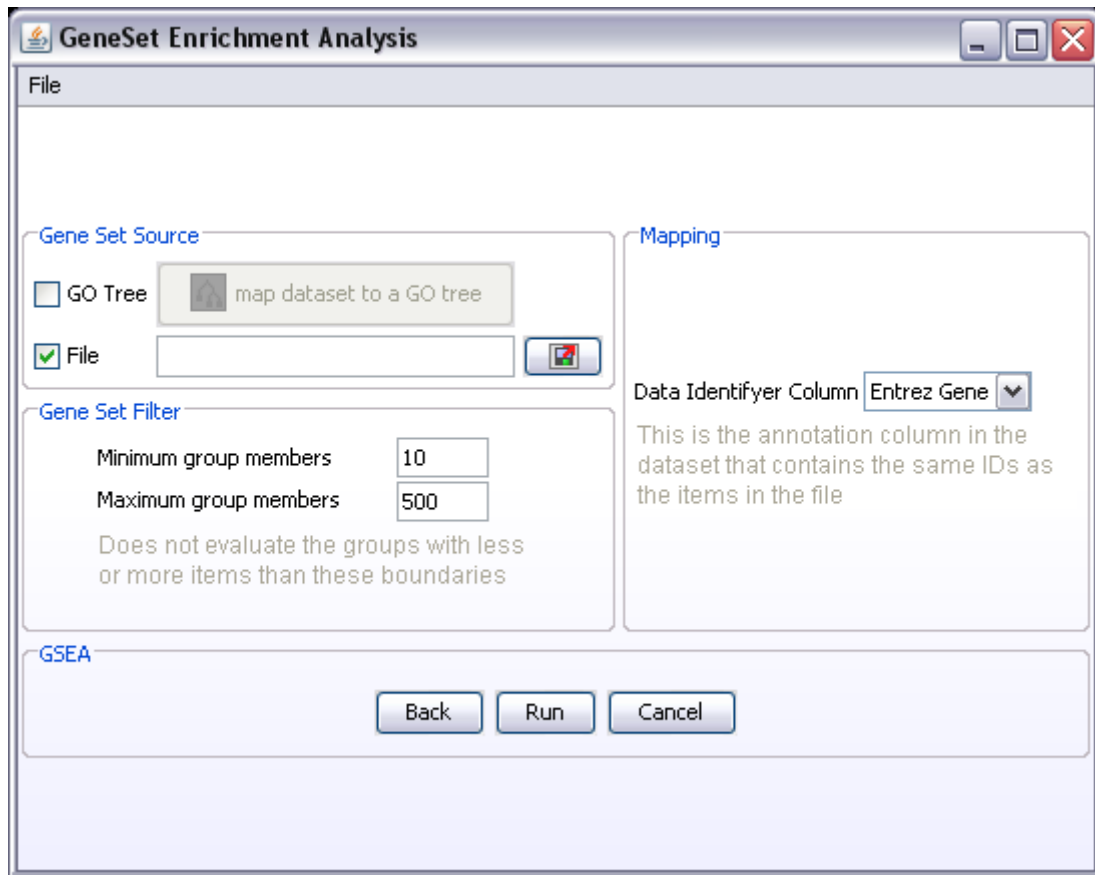
Notice that if you select File as your gene set source, the **Data Identifyer Column** becomes active. To make the connection between the dataset in j-express and the genes in the gene sets listed in the .gmt file, you must have the same id's that is used in the .gmt file available in the j-express dataset. Select this column as the Data Identifyer Column.

Gene Set Filter: Very small or very large gene sets should be filtered out. Which limits you use for minimum and maximum depends on how many genes you have in your dataset. For instance: maximum group members of 500 may be ok if you are analysing 20000 genes, but if you are only analysing 2000 genes, then genesets with 500 members may be a bit much.

Click Run. Note: before GSEA starts running the genesets will be created and the Gene Set Filters applied to remove small and large genesets. A pop-up window will appear that tells you how many genesets were created, how many were filtered and how many will be used in the analysis. You have to click ok in this window before GSEA starts.

3.24.3 Interpreting the results

An indepth explanation of interpretation of GSEA results can be found on the documentation pages of GSEA at the Broad institute.

Here follows a couple of tips on how to look at the results in J-Express.
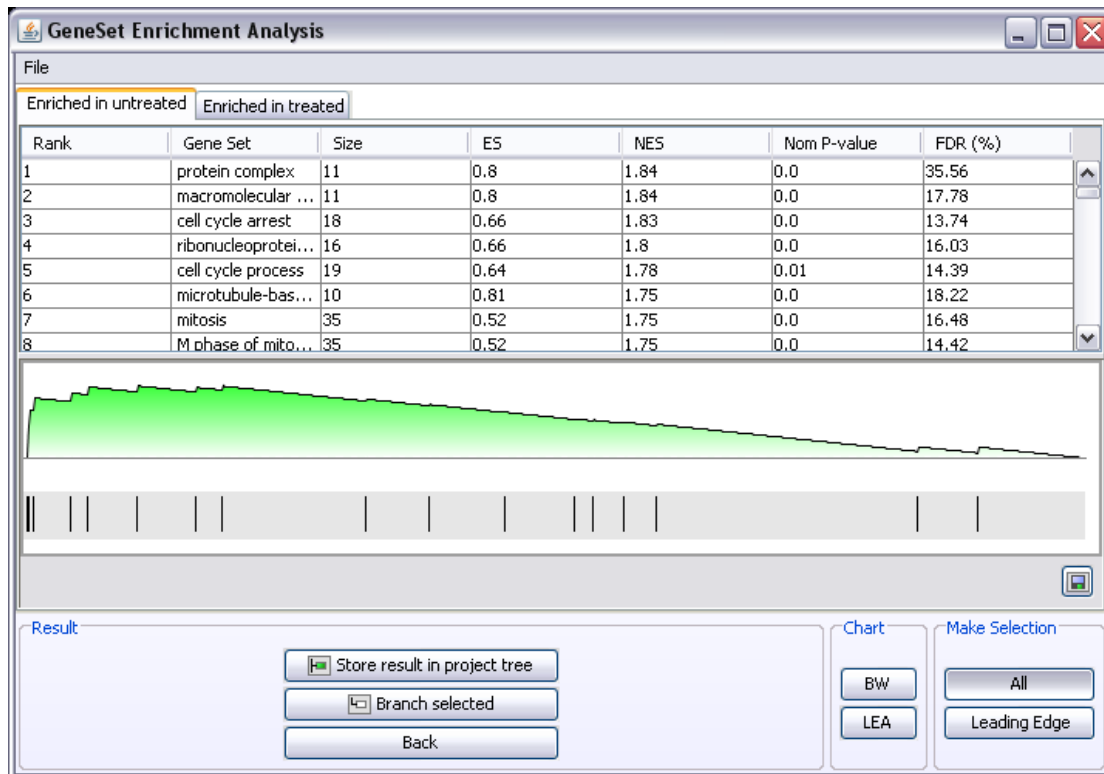
The results are presented in two tabs, one tab for the gene sets enriched in each of the groups tested. For paired analysis it will say Enriched in Group 1 and Enriched in Group 2. You can see in the Create Groups window Paired tab which samples belong to Group 1 and Group 2.

Each row in the table represent a gene set. Click on a gene set and its "random walk" will be depicted underneath. The peak of the random walk is used as the Enrichment Score (ES) for a gene set.

As usual when working in J-Express it is always a good idea to have a **Gene Graph** open next to the GSEA window. Click on the ⬛ button to open a Gene Graph of locate it under the Methods menu.

In **Gene Graph window**: click the **Shadow unselected** ▤ button.

Now when clicking on different gene sets in the GSEA window, the genes belonging to that gene set will be displayed with their gene graphs in the Gene Graph window.
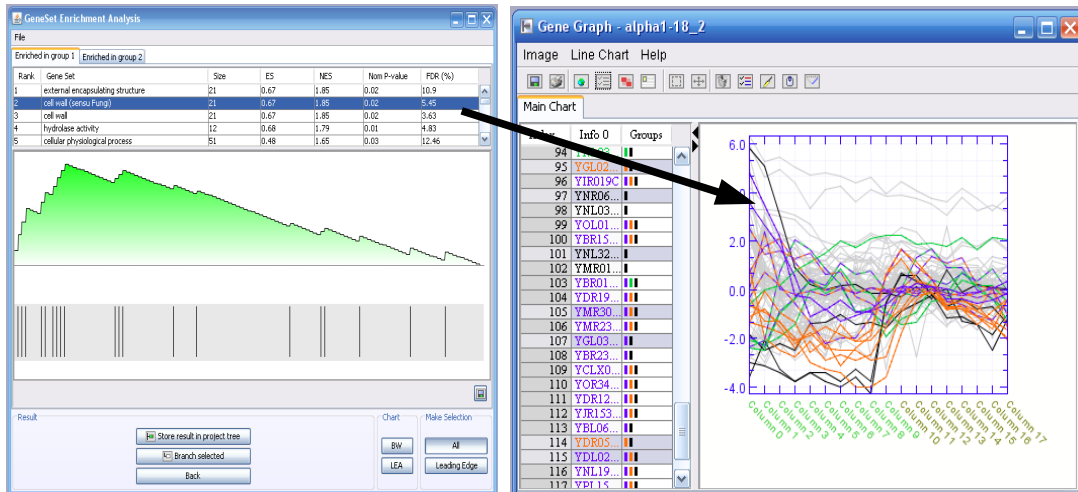
The default selection is to view all genes in the dataset that belong to a particular gene set. It is often more useful to only look at the genes that contributed to the Enrichment score. This set of genes is called **Leading Edge**. Click on the Leading Edge button in the GSEA window. Now when selecting different gene sets only the leading edge genes will be displayed in the Gene graph window.

To branch off a sub dataset containing only the genes in a particular gene set, select the gene set in the GSEA window and click the Branch button.

1. If the All button is selected, the new node in the project tree will contain all genes in the dataset belonging to the particular gene set.

2. If the Leading Edge button is selected, the new node in the project tree will contain only the leading edge genes in the dataset belonging to the particular gene set.

There are different ways of storing the results from the GSEA analysis:

1. Store result in project tree will add a GSEA result node to the project tree, which you can double-click to reopen the result window.

2. Under the file menu of the GSEA window it is possible to save the ordered list of gene sets for the selected tab including the statistical values.

3. Branch off interesting gene sets

Remember to Save the project file from the File menu in the main J-Express window.
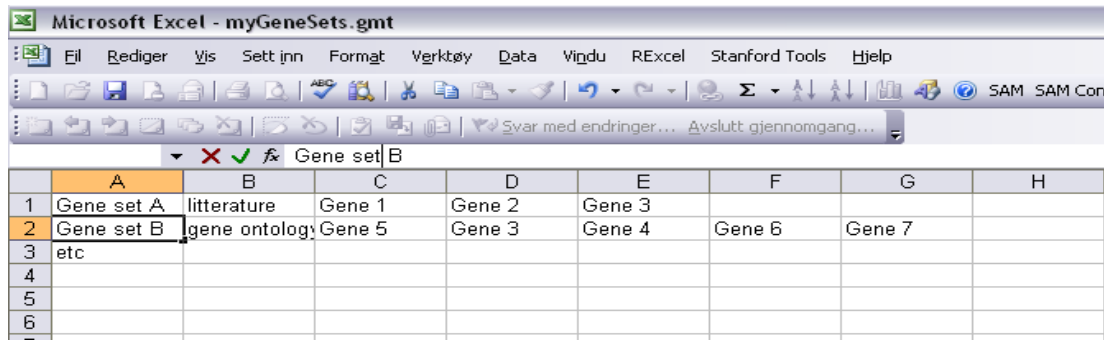
3.24.4 The .gmt file format

You can use excel to create gene sets. The format must be as follows:

Column 1: name of gene set

Column 2: Empty, or you can use this column to store information about e.g. the source of the gene set.

Column 3 and onwards: id's of gene belonging to a particular geneset

When saving the file, make sure you save the file as a tab delimited text file, and type the extension .gmt at the end of the file name. If you see in the file manager that excel has added .txt after .gmt, you can right-click the file and select Rename to remove the .txt extension.

3.25 Between Sample Fold Change
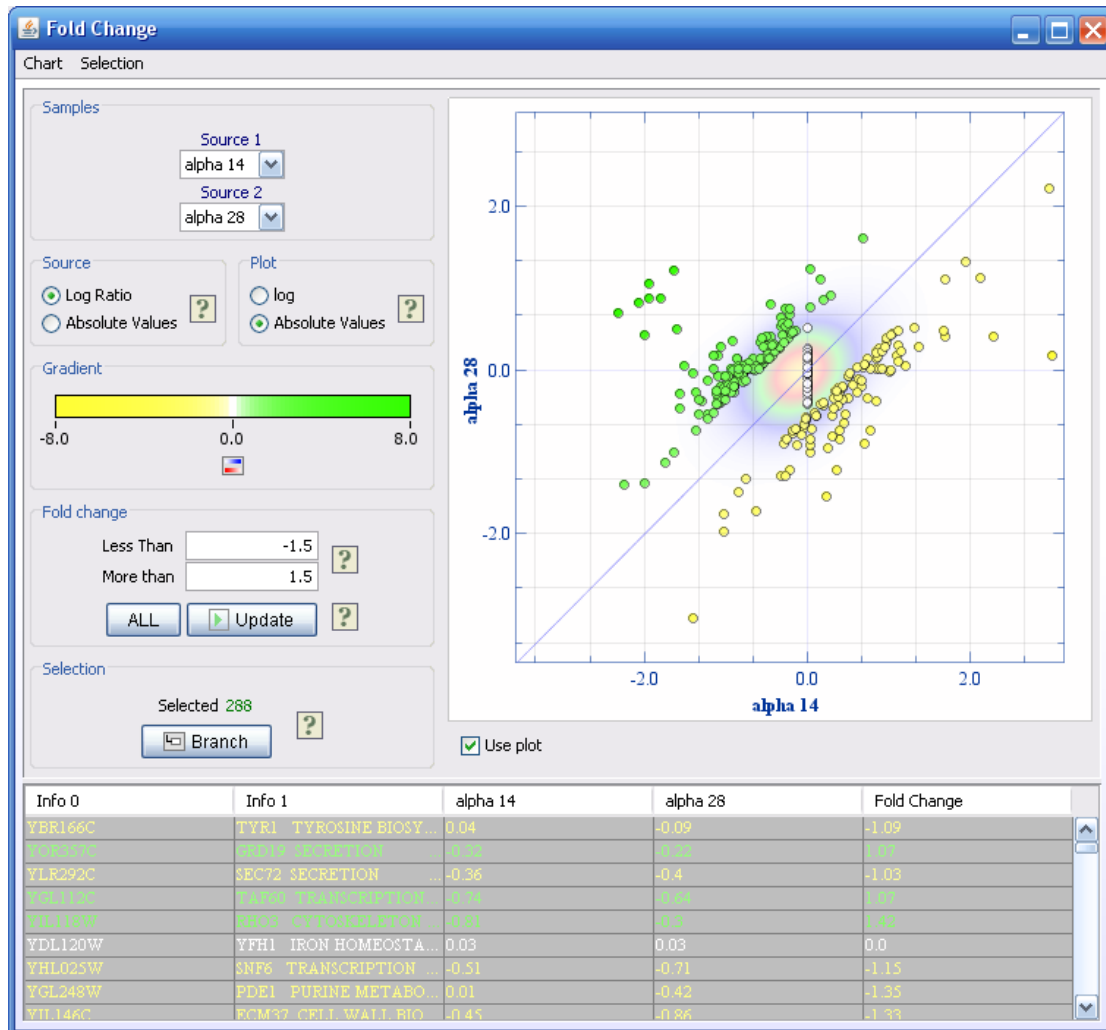
The fold change viewer can be used to see changes between two samples (columns in J-Express).

The first step in this procedure is to change the two samples to compare. Change one of the source combo boxes to get a plot of the selected samples. The calculation of fold change is different between absolute values and log-ratio values so these parameters must be set correctly before one of the sources is changes. J-Express will however try to predict the format of the data and set the parameters for source data format and plot type.

When the sources are selected and the data is plotted, you can change the plotted fold changes by either selecting rows in the table or inserting a range in the "less than" and "more than" fields and clicking update. By clicking update, all values will still show in the table, but only those between the "less than" and "more than" thresholds will be selected and shown in the scatter plot. The selection in the list are global and can also be viewed in other components such as the gene graph view (choose shadow

unselected). Also, by selecting a row in another component (such as search and sort), you can see the fold change for that selection in the scatter plot.
To change the colors for the spots in the scatter plot, click the "change color scale" button ().

By clicking Branch, you create a sub-dataset of the selection in the bottom table. This dataset will also have the correct meta information (with information about this branch).
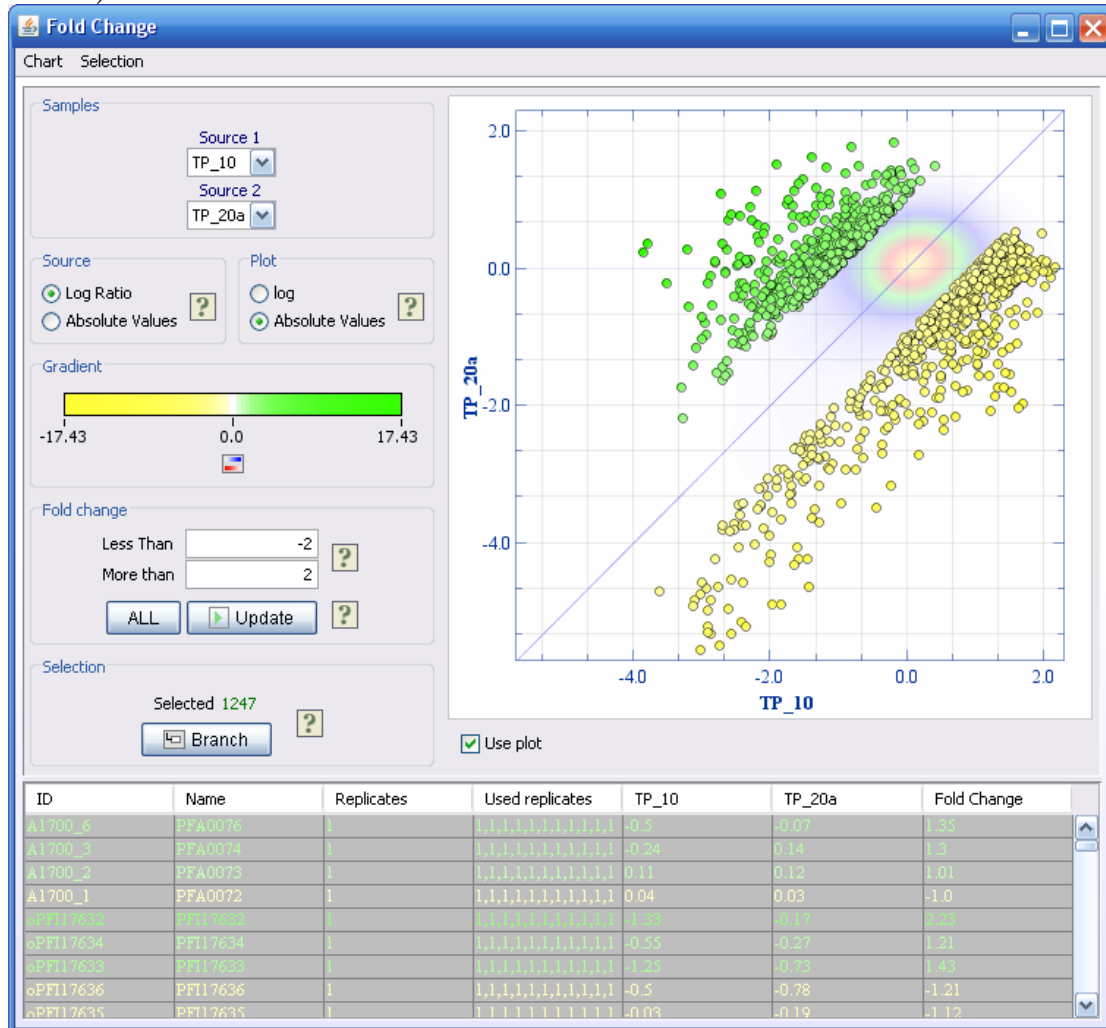
3.26 Gene Ontology Mapping

The Gene Ontology component can be used to find expression patterns from genes within a certain go-term. Using this component together with other components such as the gene graph viewer or Grouping window can effectively reveal expression changes correlated with molecular function, cellular compartment or biological process.
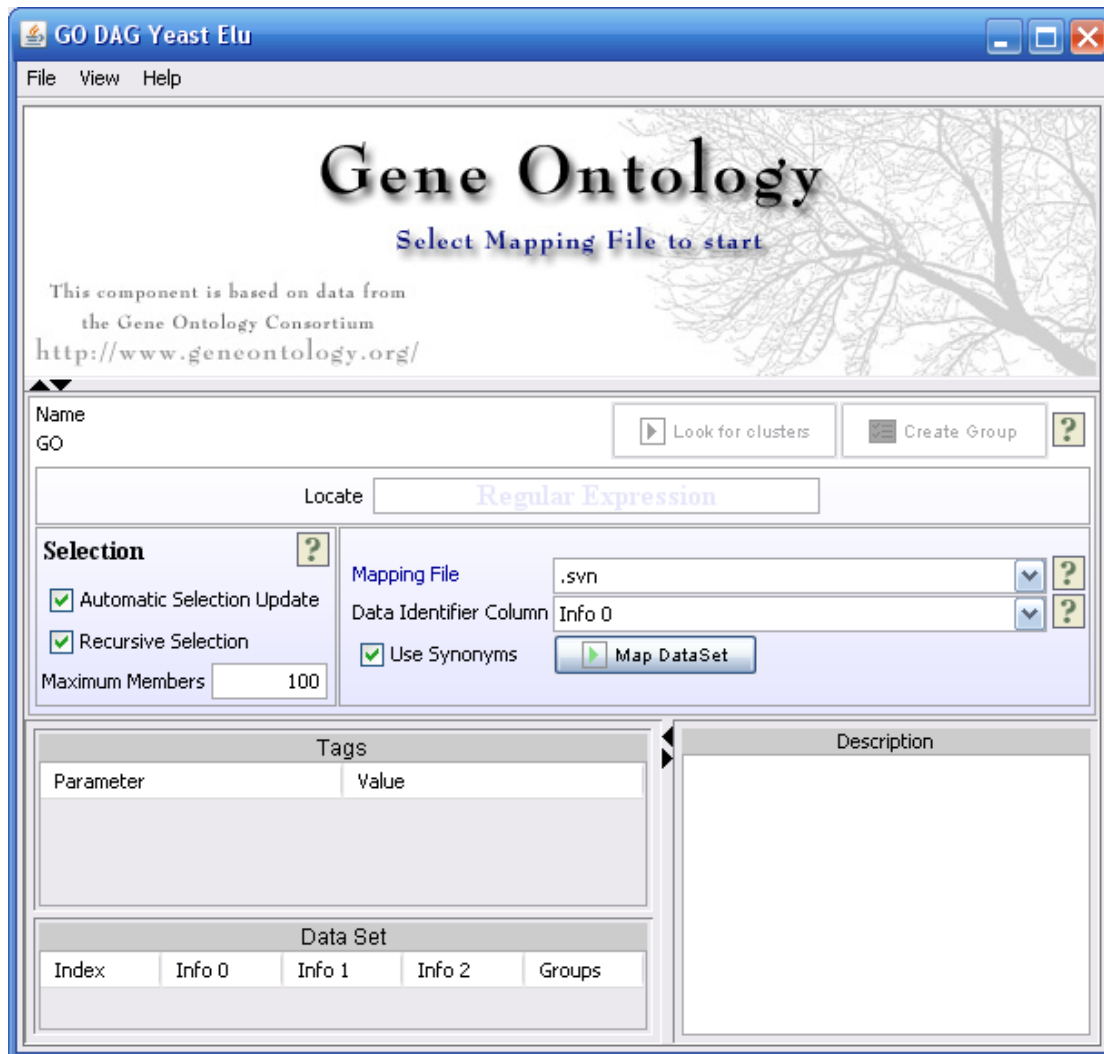
The first time this window is opened, it reads a file called gene_ontology.obo.txt. This file is located in the go folder and can be updated by downloading from http://www.geneontology.org/GO.downloads.ontology.shtml (The gene ontologies in OBO flat file format).

The next step is to select the mapping file. This file contains a mapping between certain gene identifiers and GO terms. These files must be downloaded from the Gene Ontology Consortium website at http://www.geneontology.org/GO.current.annotations.shtml and put into the folder <J-Express home directory>\ resources \go\goassociations. When selected, this file is parsed and column 3 or column 11 (if use synonyms is selected) is mapped to column

5 and then to the ontology tree. Use the *data identifier* combobox to select the identifier column in the dataset to map to the GO identifiers.

For instance, if you have a dataset with p. falciparum data and corresponding identifiers from Sanger GeneDB, you go to http://www.geneontology.org/GO.current.annotations.shtml and download the file in the row named "Sanger GeneDB Plasmodium falciparum". When downloaded, you put this file in the folder called c:\program files\Molmine AS\J-Express Pro 2.x\resources\go\goassociations. Then click the Gene Ontology button or select from the methods menu. Select the file *gene_association.GeneDB_Pfalciparum.gz* in the mapping file box. You can then browse the tree or look for clusters.

Selection

The selection frame let you create dataset selection based on selection of the GO-terms in the tree.

- Automatic Selection Update creates a dataset selection when a GO-term is selected. This selection can be viewed in for instance the gene graph component (using *shadow unselected*).
- Recursive Selection selects all genes in a selected GO-term and includes GO-terms in other tree-nodes downwards in the GO-tree.
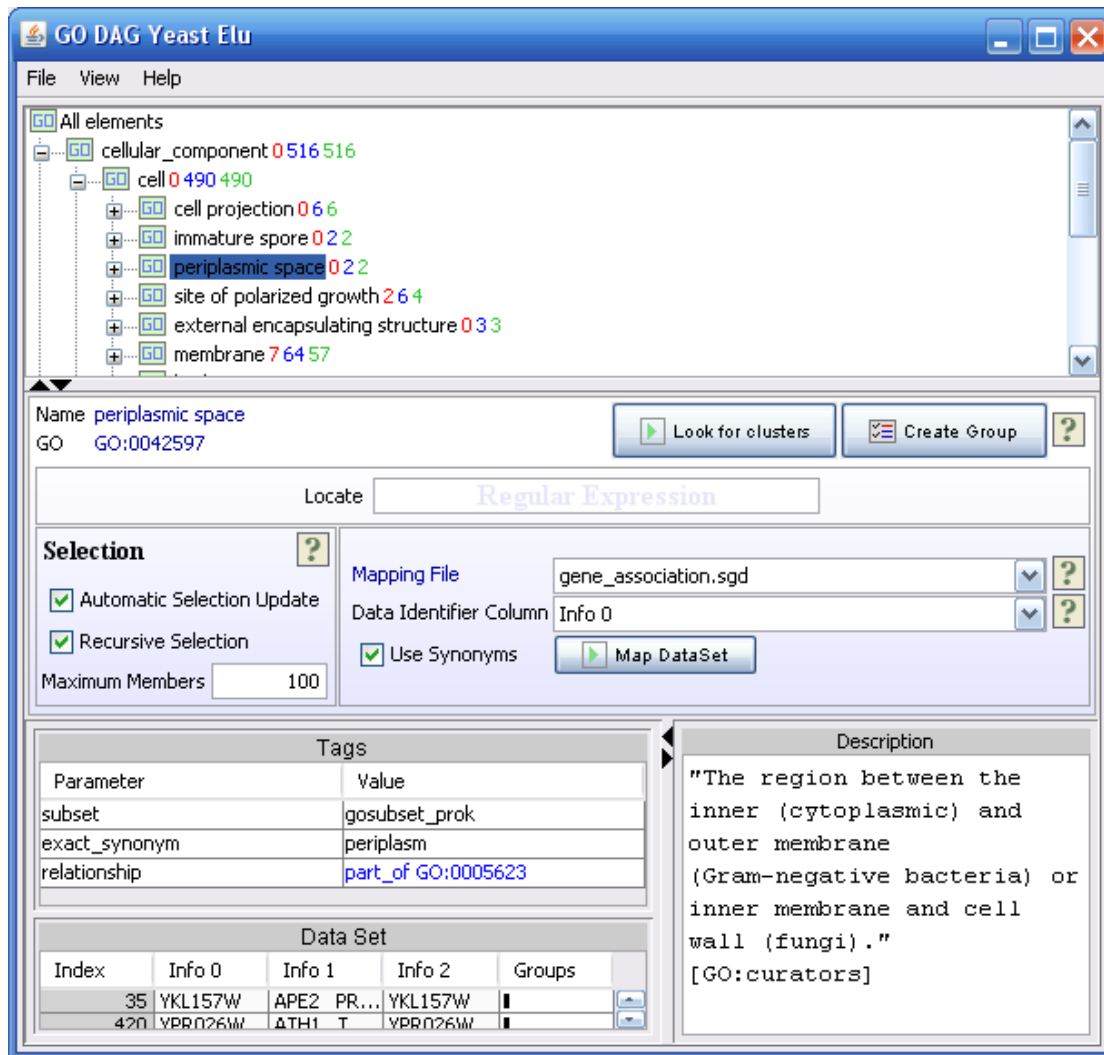
The figure shows a GO-tree mapped to a yeast dataset. The red numbers in each term shows the number of genes in the selected dataset that corresponds to this GO-term. The blue numbers is the total number of genes corresponding to this GO-term and other terms downwards in the tree (child terms).

If both the red numbers and blue numbers are 0 for all terms, you are probably searching with the wrong identifiers. If this is the case, you should download a different mapping file (see above) or try to create a new column of compatible identifiers using for instance the ID linker.

The tags table shows all information tags contained in the selected GO-term. The DataSet table shows you all the genes in the selected dataset corresponding to this GO-term. This table depends on the selection in the Selection frame. Selecting genes in the DataSet table will fire a global selection event that can be viewed in a gene graph viewer or a grouping dialog.

The definition window shows you all information about the selected GO-term.

Look for Clusters

This button opens the cluster window that enables searching for clusters within the GO-terms. By clicking on a row, the selected GO-term will also open in the GO-tree. Select a Method for generating clusters and click *find clusters*.

3.27 Selection Viewer

The selection Viewer is simply a window showing all selected indices for a dataset. As you select between the active datasets in the project tree, it can be hard to keep track of the genes that are selected for each dataset. This selection can for instance be viewed in a Gene Graph window by choosing shadow unselected. You can also use the grouping component to create a group of the selected genes.

3.28 Selection Chart
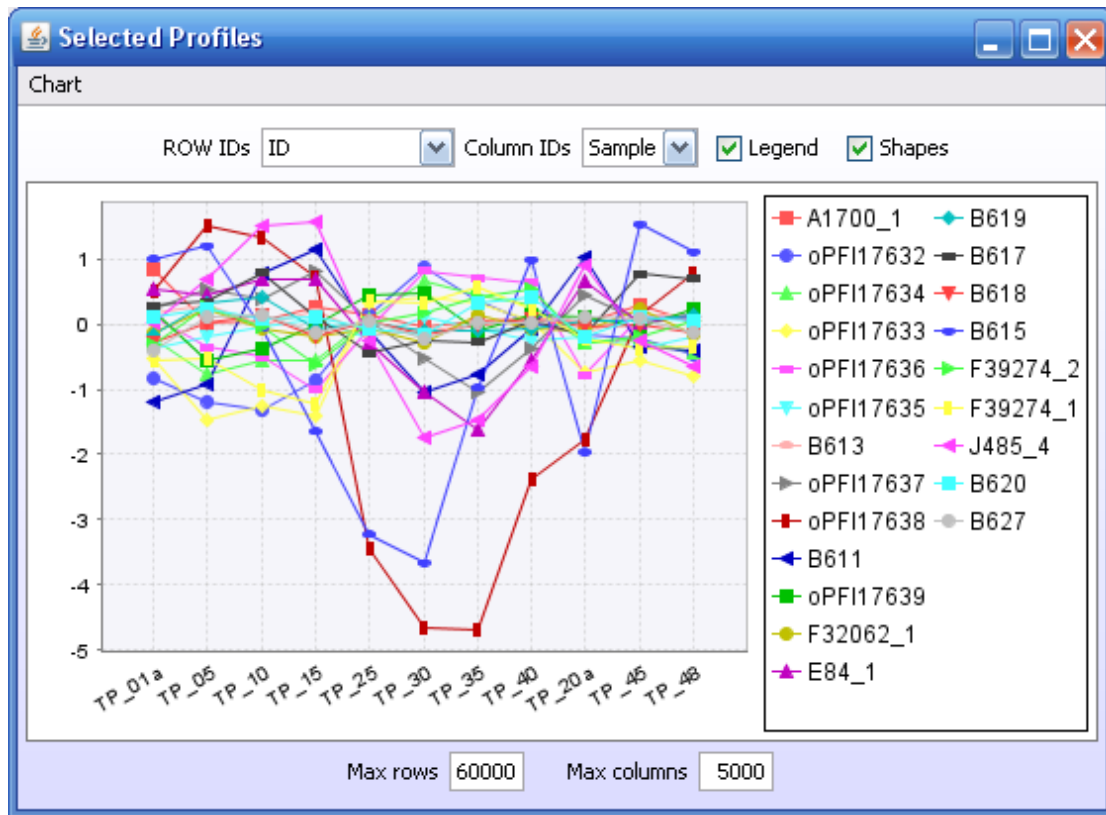
The selection Chart is simply a window showing all selected profiles for a dataset. To see the selected genes in a chart, just keep this window open. To prevent locking J-Express when drawing thousands of genes, the max rows and max columns prevent too large datasets from showing too much data. If there are multiple annotation rows or columns, select the annotation to include with the top combo boxes.

3.29 Scripting

With the scripting feature of J-Express Pro, you can automate your microarray data analysis and create new methods and visualizations. The script interface can help you save time when performing repetitive operations on your dataset(s). You have access to the data objects, and may manipulate your data matrices by using some of J-Express' native methods, or manipulate completely on your own. You can also use the script interface to connect data from J-Express with your own java classes.

Two different types of scripting exist in J-Express from version 2.7:

Jython script (a hybrid between python and java)
Java script

Jython scripting enables full support for python, but also lets you use java objects directly in the scripts. **It is strongly encouraged to use this type of scripts** as the java script can be quite difficult to use. See example scripts for code examples and use the scripting forum at www.molmineus.com/forum for support and help.

Jython scripting is also available for low-level data preparation and enables scripting in the array processing step. See examples.

Information about the JavaScript standard and additional documentation can be found at the web site: http://www.mozilla.org/rhino/doc.html

Internal J-Express Objects
Some classes exist in J-Express that gives access to various framework objects such as the internal desktop and the project tree. Most structures used in J-Express are available from the **main** object. The other central object is the **data** object which contains all available information about the selected dataset.

Here are the most important fields in the two framework classes and the Group class for group information:

Main (the main class)

Fileld	Type	Description
MW.JDesktopPane1	JDesktopPane	The main desktoppane containing all InternalFrames
addNode(TreeNode child,TreeNode parent)	void	Adds a project node (for instance a dataset) to the parent node
getProjectTree()	JTree	The main project tree
getProperties()	Hashtable	Settings that are saved when J-Express closes and loaded at startup. Remember that objects put into this table must be serializable.
getSelected()	Object	The selected object (node) in the

		project tree. Are usually casted to DataSet (This is done automatically in jython: for instance Dat = main.getSelected() where Dat is used a DataSet object)
getTreeModel()	DefaultTreeModel	The project tree TreeModel
getTreeRoot()	TreeNode (DataSet)	The root of the project tree
numFormat	java.text.DecimalFormat	The decimalformatter used in all objects

Example:
#Create a JInternalFrame and put it into the main desktop
dialog = JInternalFrame("Script")
dialog.getContentPane().add(new JLabel("test"))
dialog.setLocation(300,300)
dialog.pack()
dialog.setClosable(1)
dialog.setResizable(1)
dialog.setIconifiable(1)

main.MW.JDesktopPane1.add(dialog)
dialog.setVisible(1)
#store some values in the Hashtable storage
Hashtable hash = main.getProperties()
hash.put("aninteger",new Integer(3))
hash.put("aString","this is a string")

#Next time J-Express starts, the following script is valid:
Hashtable hash = main.getProperties()
Integer aninteger = hash.get("aninteger")
aString astr = hash.get("aString")
print astr

Data (The data containing class, always the dataset selected in the project tree)

FileId	Type	Description
DataSet(double[][] data,String[][] infos,String[][] colinfos)	Creator	Creates a new dataset from a double[][] array of data, a String[][] array of row(gene) annotation and a String[][] array of column annotation.
addColumnGroup(Group gr,boolean last)	void	Add a group of samples (columns) to the dataset, see

		description for the Group Object below
addGroup(Group gr,boolean last)	void	Add a group of genes (rows) to the dataset, see description for the Group Object below
extract(Vector members)	DataSet	The main method for creating subsets of data. The members Vector should contain Integers where each integer is a row that should be in the result dataset. This dataset is initially linked (contains only pointers to data from the parent dataset). To unlink the dataset (create its own *data[][]* vector) call the *setParentDataSet(DataSet parent)* first and then *unLink(0)*. To connect it to a dataset int the project tree, call *main.addNode(TreeNode child,TreeNode parent)*
extractColumns(Vector members)	DataSet	Same as above for columns.
fireSelectionChangeEvent (Object source)	void	Fires a change event so that all listener listening for selection changes are updated. Use this together with *setSelectedRows(int[] selectedRows)* or *setSelectedColumns(int[] selectedCols)*
getColInfos()	String[][]	Annotation for all samples
getColumnGroups()	Vector of Group objects	All sample groups (see Group object)
getData()	double[]	The actual expression

		[]	matrix
getDataLength()		int	Number of rows (genes) in the dataset
getDataWidth()		int	Number of Columns (Samples) in the dataset
getFile()		String	The name of the dataset (appears in the project tree)
getGroups()		Vector of Group objects	All gene groups (see Group object)
getIconImage()		ImageIcon	The icon of the dataset
getInfo()		String	The info field
getInfoHeaders()		String[]	Row annotation headers
getInfos()		String[][]	The gene (row) annotation
getSelectedColumns()		int[]	The column selection
getSelectedRows()		int[]	The row selection
getStructures()		Hashtable	A hastable to store anything together with the dataset. Remember that object in this hash must be serializable
getnulls()		boolean[][]	The missing values in this dataset
hasNaN()		boolean	True if there is any NaN values in the data
linked()		boolean	True if this dataset does not contain data of it's own, but only has indices to the parent dataset
reLink(boolean showWarnings)		void	Removes data and links to the parent dataset
setColInfoHeaders(String[] colinfoHeaders)		Void	Set the headers for column annotation
setColInfos(String[][] colinfos)		Void	Set column annotation
setColumnGroups(Vector classes)		Void	Reset all column groups (Vector of Group)
setData(double[][] data)		Void	Set the data
setFile(String file)		Void	File is the name of the

		dataset
setGroups(Vector classes)	Void	Reset all row (gene) groups
setIcon(ImageIcon icn)	Void	Set the icon for this dataset
setInfo(String info)	Void	Set the info field
setInfoHeaders(String[] headers)	Void	Set headers for row (gene) annotation
setInfos(String[][] infos)	Void	Set row (gene) annotation
setMetaList(expresscomponents.Documentation.Meta InfoList MetaList)	Void	Meta list is the list of meta info
setStructures(Hashtable structures)	Void	Structures is a hashtable that is saved with the dataset. It can be used to store any kind of serializable object
setnulls(boolean[][] nulls)	Void	Set a matrix of missing values (true represents a missing value)
unLink(boolean showWarnings)	Void	Unlink the dataset from its parent dataset. This will copy all data overlapping between this dataset and its parent to this dataset.

Group (the grouping information, both on rows and columns (see DataSet))

Fileld	Type	Description
Group()	Creator	Creates a new empty group
Group(boolean active, String name,Color color,boolean[] members, LineMark lineMark, String description)	Creator	Creates a new group. The members are representing the row or columns included in this group. This number must be the same dimension as the number of rows or columns. DataSet.addGroup(Group group, Boolean last) or DataSet. addColumnGroup(Group group, Boolean last) can be used to add the group to a dataset.
setActive(boolean active)	Void	Turn this group on or of
getColor()	Color	The group Color
setColor(Color color)	Void	Set the group Color

setDescription(String description)	Void	Set the group description
setMembers(boolean[] members)	Void	The rows or columns that should be a member of this group.
setName(java.lang.String name)	Void	Set the name of the group
getCopy()	Group	Get a clone of this group
getDescription()	String	Get the description n of this group
getGroupCount()	String	The number of members in this group
getMembers()	boolean[]	The group members
getName()	String	The name of the group
isActive()	boolean	True if this group is active
isMember(int row)	boolean	True if "row" is a member of this group

Importing objects

Objects that are already in the J-Express classpath can be created by including their package in an import statement like:

```
from myclasses import myclass
```

Adding new libraries

By putting a jar-file in the J-Express lib-folder, the library will automatically be included in the class path. Objects can then be created by including them in import statements.

Example:

A library called mylib.jar has a class called myclass in package myclasses with a constructor myclass(String str) and method String mymethod(int anint).
We can use the library by putting it into the J-Express lib-folder. The following script is then valid:

```
from myclasses import myclass

aclass = myclass("a string")
anotherstring = aclass(55)

print anotherstring
```

For instance, the jFreeChart library is already present in the J-Express lib folder. New charts can be generated in the following way (for complete reference of the JFreechart API, please refer to http://www.jfree.org/jfreechart/javadoc):

This script calculates the 3 first principal components using the Jama library and plots them in a JFreechart line chart. (The script is available as an example script in the J-Express script folder).

```
from java.lang import *
from org.jfree.chart import *
from org.jfree.chart.plot import *
from Jama import *
from expresscomponents import JDoubleSorter
from org.jfree.chart.renderer.category import *
from org.jfree.chart.axis import *
from org.jfree.data.category import *
from java.awt import Rectangle
from javax.swing import JInternalFrame

sel = [4,5,6,7,90,12,44,43,43,22,11]
dat = data.getData()
m=data.getDataWidth()

colinfos = data.getColInfos()
rowinfos = data.getInfos()

A = Matrix(dat);
SVD =SingularValueDecomposition(A)

R = SVD.getSingularValues()

m2 = len(R)

dataset = DefaultCategoryDataset();

M=SVD.getV()

ARR = M.getArray()
r1 = M.getRowDimension()
c1 = M.getColumnDimension()

princ1=[]

#This is the principal component containing most of the variance
for j in range(0,m):
#princ1=princ1+[ARR[j][0]]
princ1=princ1+[ARR[j][2]]
dataset.addValue(ARR[j][0],String("PC1"),String(String.valueOf(j)))
dataset.addValue(ARR[j][1],String("PC2"),String(String.valueOf(j)))
dataset.addValue(ARR[j][2],String("PC3"),String(String.valueOf(j)))

chr = ChartFactory.createLineChart( "", "", "", dataset,
PlotOrientation.VERTICAL, 1,0,0 );
```

```
plot = chr.getCategoryPlot()

domainAxis = plot.getDomainAxis()
domainAxis.setCategoryLabelPositions(
CategoryLabelPositions.createUpRotationLabelPositions(Math.PI / 6.0)
);

data.setSelectedRows(sel)
data.fireSelectionChangeEvent(data)

rend = plot.getRenderer()
rend.setShapesVisible(1);

#Create the panel
panel = ChartPanel(chr)

#Create the dialog frame
dialog = JInternalFrame("Script")
dialog.getContentPane().add(panel)
dialog.setLocation(300,300)
dialog.pack()
dialog.setClosable(1)
dialog.setResizable(1)
dialog.setIconifiable(1)

main.MW.JDesktopPane1.add(dialog)
dialog.setVisible(1)
```

3.29.1 Basics about the java script interface.

The script interface uses a JavaScript interpreter to execute the scripts. So the
language the scripts must be written in is JavaScript. You can use all the native
JavaScript features, and use them to manipulate your data.

Using the script interface in combination with J-Express' own classes will eliminate a
lot of "clicking" on buttons, parameter settings, and move these operations into the
script. The script can be saved and re-loaded. So if you wish to repeat an earlier
performed task, simply reload the script, and run it.

The way of accessing a DataSet in the project tre, is to select it. Then it will be
available in the script window as the variable "active". When you wish to perform
scripting on another dataset, simply select another dataset, and the "active" variable
will be updated.

When initializing objects in JavaScript, it is necessary to write "Package" prior to the full package name of the object you are initializing. An exception is when you are using native java classes like:

java.lang.System.out.println("Hello");
new java.util.Vector();

But when using for example a J-Express class:

new Packages.expresscomponents.Scripting.Launch(master,0);

An alternative to writing the package name, for example if you use the same class several times, is to use the *importPackage(Packages.etc....)* or the *importClass(Packages.etc.....MyClass)* statement.

3.29.2 The Examples - getting started.

Some useful examples are included, covering normalization of one and two channel data, standard task like different clustering methods etc.

3.29.3 The class expresscomponents.Scripting.Launch

The "Launch" class - as the name implies is a "helping" class to initialize the most common features. The methods are simply wraps some of the native J-Express class constructors into a collection. The constructor takes as parameter the main window, in the script interface called *master* and an integer number, referring to the distance measure to be used by the launch object.

In general, the Launch class has two methods for each of the main procedures in J-Express. One that only takes a single DataSet variable as input, and returns the object in question, for example a pca analysis. The second method is a more comprehensive that takes as input all the different settings that you would have been able to set in the settings window.

Please inspect the API of the Launch class for further information.

3.29.4 Using J-Express classes directly.

All the java classes in J-Express can be used from the script interface. Even though, some of the classes are not very well suited for use with the script interface. A more script-friendly version of the J-Express API will be released later.

The class expresscomponents.DataSet is the class that contains all information about the data, and has several methods for manipulating the data.

An example:

subsetVector=new java.lang.Vector(10);
subsetVector.add(3);
subsetVector.add(55);
.
.
.
.
newDataSet=active.extract(subsetVector);

pca = new Packages.jexpress.mainPCA2(master, newDataSet);

These lines of code does the following: First note that the variable *master* and *active* are accessible when you start the script-interface. The *master* variable is a reference to the main window of J-Express, and *active* is a reference to the dataset that is selected in the project tree.

First, a vector is initialized. Then som integers are added (3, 55, ...). They correspond to indices in the DataSet. The *newDataSet* - a new dataset consisting only of the rows (3, 55, ...) in the original *active* dataset. Last, a new *mainPCA2* object is initialized, with the new dataset as parameter.

This will show a standard PCA window of the new dataset.

Using the Launch class instead, the code would be:

launcher=new Packages.expresscomponents.Scripting.Launch(master, 0);
subsetVector=new java.lang.Vector(10);
subsetVector.add(3);
subsetVector.add(55);
.
.
.
.
newDataSet=active.extract(subsetVector);

pca = launcher.newPCA(newDataSet);

Note that when using the Launch class, the master object is only used when the Launch object is initialized.

Similar examples are included in the example scripts.

3.30 DataSet repository

The J-Express DataSet server and client framework has been replaced with a DataSet repository.

The JExpress client can access a server-based repository to store complete datasets. Stored data includes raw data and associated information (description, processes, meta-data etc). This information can then be shared with other users. Unauthorized access is prevented by placing each dataset in a folder which only specific groups have access to.

Corporate customers can set up their private repository server, see last section for details regarding this.

All JExpress registered users can gain free access to the Molmine public repository. To to this, users must register their account (see next section). After registration, the users can save their data to the server, share it with colleagues and co-researches, and access those data from every computer connected to the Internet. The server url of the public repository is http://katsura.bccs.uib.no:8088/molmine/ (which is the default when you start the repository browser for the first time).

Disclaimer: The Molmine server has a quota limit of 500MB per user, 1 user pr person. Note that Molmine is not responsible for the security or safety of the data you choose to save to the Molmine respository.

3.30.1 Starting the repository browser and registering an account

To start the repository browser, choose "Browse server repository" from the "File" menu, or press Ctrl-Q.

The window that opens initially have only one tab called "Server". The first step is to create your account on the Molmine server. Select the "New user" box, choose a user name and password, fill inn "Full name" and click "Create new user". The user account is then created. For security reasons, initially the user account will be marked as "Inactive". It will then be activated by the repository administrators. When the account is activated, you are ready to login (see next section).

3.30.2 Server settings and logging in

Start the repository browser by pressing Ctrl-Q. Leave the field "Server url" unchanged unless you are using another repository and have that address provided to you by the repository administrator. Fill in user name and password and click "Log in". If the login is successful, another tab called "Data sets" should be opened. The last used server url, user name and password is stored for your convenience.

3.30.3 Viewing and editing datasets and folders

The folder list on the left will show you a tree view of the folders you have access to. Double-click on the folders to expand them. Click on an data set to view information on that particular dataset, including description, who submitted it, date of submition etc. If you want to load that particular data set, click "Load". Note that this may take some time, depending on the size of the data set. The size of the dataset is shown (in megabytes) in the details panel.

If you want to move a dataset from one folder to another, you can do so by dragging them in the folder tree: Press the left mouse button on the dataset you want to reorganize, move the mouse pointer to the new folder and release the mouse button.

You can edit the name of the data set and the description. After finishing editing, click "Update dataset details" to save your changes.

To change the name of a folder, click on the name and wait a second or two. You can then edit the folder name. Hit ENTER when finished (otherwise your editing will be lost).

If a folder is empty, you can delete it by choosing "Delete" from the right-click menu. To delete a non-empty folder, first delete or move all data sets and subfolders.

To create a new folder, right click on the folder in which you want to place the new folder. Choose "New folder" from the right-click menu. After the folder has been created you can rename it to a more descriptive name.

Note: If you know a data set has been uploaed but you are unable to find it, it can be in a folder that you don't have access to. Contact the repository administrator to get

the necessary permissions, or tell the person who uploaded that data set to move it to a folder you have access to.

3.30.4 Saving new datasets to a repository

To save a data set that you have loaded into the JExpress client, right-click on the data set node in the project tree and choose "Copy dataset". Then go to the repository browser, find the correct folder in which to place the data, and select "Save dataset" from the right-click menu. Note that this may take some time, depending on the size of the data set.

3.30.5 Trouble shooting: Network settings and firewalls

The JExpress client must be able to initiate outgoing HTTP connections on port 8088. If the client is unable to connect, verify that this port is open in your network by opening the server url (e.g. http://katsura.bccs.uib.no:8088/molmine/) in your web browser. Contact your network administrator if you are unable to resolve the issue.

3.30.6 Setting up a dedicated server

By special agreement only. Contact Molmine for details.

3.31 Plugins

3.31.1 Creating Plugins

J-Express polugins no longer need to be subclasses of the plugin classes. Instead, they are initiated from a jython script. All jar-files placed in the J-Express plugins folder at startup will be included in the classpath. Starting the plugin must be done by passing the correct parameters to your plugin class from the script interface. A couple of test scripts are included in the main J-Express installation.

A plugin becomes available from the J-Express framework when an XML file with the below parameters are found either in the plugins folder or within a jar file located in the plugins folder.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<java version="1.5.0_06" class="java.beans.XMLDecoder">
 <object class="expresscomponents.plugins.Settings">
  <void property="buttonImage">
   <string>/plugins/stat1.gif</string>
  </void>
  <void property="launchScript">
   <string>

from javax.swing import JOptionPane
from plugins import *

if data==None or data.getDataLength()==0:
        JOptionPane.showMessageDialog(main.MW, "No Data Set
Selected", "Missing Data", JOptionPane.ERROR_MESSAGE)
else:
        st = Stat1(main,data)

</string>
  </void>
  <void property="pluginName">
   <string>Statistics tutorial Plugin</string>
  </void>
<void property="Description">
   <string>A Statistics tutorial Plugin for calculating t-score and
regularized t-score</string>
  </void>
 </object>
</java>
```

The script is launched whenever a plugin is started by clicking the plugin button or menu in J-Express. "/plugins/stat1.gif" is the icon of the plugin and is in this case located in a jar-file.
Stat1 is the plugin class and is in this case started with Stat1(main,data).
If this plugin was an internal frame it could be added to the main J-Express frame with for instance:

```
main.MW.JDesktopPane1.add(st)
st.setVisible(1)
```

For more examples, see the plugins folder.

Join the J.Express forum at www.molmine.com/forum to share your scripts or plugins with the J-Express community or ask questions.

4 Method and Algorithm Description

4.1 Distance measures

Some of the methods described in this chapter take a high dimensional data matrix as input and give the results as a "rearrangement" of this input. The rearrangement is usually performed with regard to some similarity or dissimilarity measure. In clustering algorithms, two relatively similar objects should be placed in the same cluster. In projection algorithms, they should be placed in the vicinity of each other in the projected space.

Input to our algorithms is an n x m dimensional matrix with n vectors of length m. We assume that all vectors are of the same length. When we refer to vector x in state k, it is generally element (x,k) in the input matrix. We refer to the distance between vector x and vector y as d(x,y).

If the input matrix consists of n vectors in 1 state (nx1 dimensional input matrix) the similarity decision is simple: $d(x,y)=|x_1-y_1|$. If vector x has a value of 3.0 and vector y has a value of 1.0 the distance between them is d(x,y)=3.0 − 1.0 = 2.0. The distance from x to y should be the same as the distance from y to x (symmetric distance), so we define a mathematical equation for the distance **(1)**:

Distance in one dimesional space.
$$d(x, y) = \sqrt{(x_1 - y_1)^2} \qquad \textbf{(1)}$$

Some frequently used distance functions.	
Camberra : $$d(x,y) = \sum_{i=1}^{m} \frac{\left|x_i - y_i\right|}{\left|x_i + y_i\right|} \qquad \textbf{(2)}$$	Euclidean : $$d(x,y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2} \qquad \textbf{(5)}$$
Minkowsky : $$d(x,y) = \left(\sum_{i=1}^{m} \left|x_i - y_i\right|^r \right)^{1/r} \qquad \textbf{(3)}$$	Manhattan / city - block : $$d(x,y) = \sum_{i=1}^{m} \left|x_i - y_i\right| \qquad \textbf{(6)}$$
Chebychev : $$d(x, y) = \max_{i=1}^{m} \left|x_i - y_i\right| \qquad \textbf{(4)}$$	

The most commonly used proximity measure, at least for ratio scales (scales with an absolute 0) is the Minkowski metric **(3)**, which is a generalization of the distance between points in Euclidean space.

The following is a list of the common Minkowski distances for specific values of r:

r = 1. Manhattan /City block distance. A common example of this is the Hamming distance, which is just the number of bits that are different between two binary vectors.

r = 2. Euclidean distance. The most common measure of the distance between two points.

r →∞. "supremum" (L_{MAX} norm, L_∞ norm) distance. This is the maximum difference between any component of the vectors.

The distance functions implemented in J-Express:

Euclidean	$d(x,y) = \sqrt{\sum (x_i - y_i)^2}$
Squared Euclidean	$d(x,y) = \sum (x_i - y_i)^2$
Manhattan	$d(x,y) = \sum |(x_i - y_i)|$
Canberra	$d(x,y) = \sum \dfrac{|x_i - y_i|}{|x_i + y_i|}$
Chebychev	$d(x,y) = \max(|x_i - y_i|)$
Bray Curtis	$d(x,y) = \dfrac{\sum |x_i - y_i|}{\sum x_i + y_i}$
Cosine Correlation	$d(x,y) = \dfrac{\sum (x_i y_i)}{\sqrt{\sum (x_i)^2 \sum (y_i)^2}}$
Pearson Correlation	$d(x,y) = \dfrac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (y_i - \bar{y})^2}\sqrt{\sum (y_i - \bar{y})^2}}$
Uncentered Peason Correlation	$d(x,y) = \dfrac{\sum x_i y_i}{\sqrt{\sum (y_i - \bar{y})^2}\sqrt{\sum (y_i - \bar{y})^2}}$
Euclidean Nullweighted	Same as Euclidean, but only the indexes where both x and y have a value (not NULL) are used, and the result is weighted by the number of values calculated. Nulls must be replaced by the missing value calculator (in dataloader).

A weakness of the standard Minkowsky distance measure is that if one of the input attributes has a relatively large range, then it can overpower the other attributes. For example, if x_0 has a value range of 0-100, and x_1 has a value range from 0 to 10, then x_0's

influence on the distance function will usually be overpowered by x_1's influence. This is normally not a problem in microarray experiments, as all attributes generally have the same value span. However, to prevent overpowering, the data is often normalized. A simple way of normalizing the vectors is variance and mean normalization, (EQN 7).

Variance and mean normalization
$$Z_i = \frac{(x_i - \mu_i)}{\sigma_i} \qquad \text{(7)}$$

It is very important to understand the effect that normalization has on the data. In the case of equation **(7)**, vectors with very small absolute values will be scaled to have the same variation as vectors with initially very large values (4.1). Sometimes this is not desirable.

Figure 1. The effect of variance and mean normalization

Before normalization

After normalization
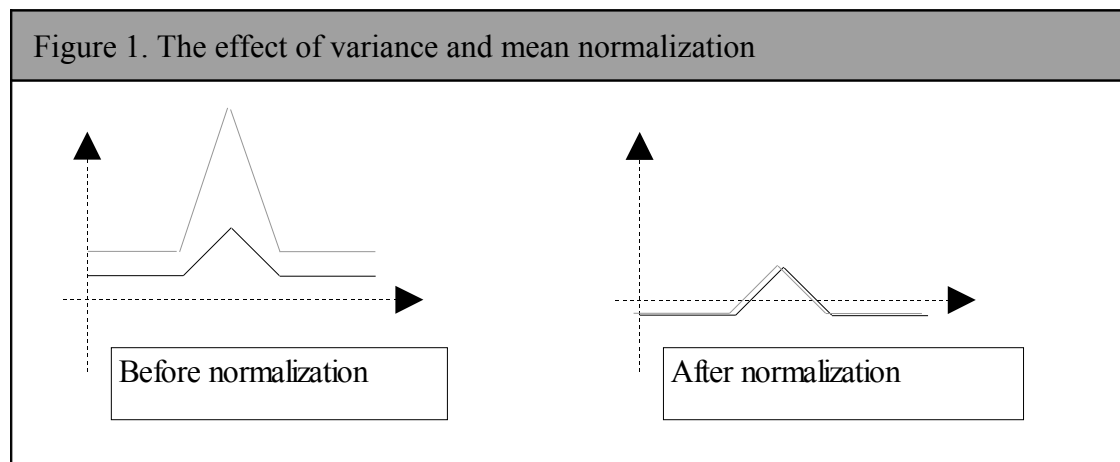
If we are just looking for profile similarity, that is the shape of the lines, normalization prior to distance calculation is appropriate and allows a simple distance measure (e.g Euclidean) to be used. However, if the absolute value of the vector has some meaning, this will be lost after variance normalization.

Typically, the different distance measures falls into one of two classes: metric and semi-metric. To be classified as 'metric', a distance between two vectors x and y must obey several rules:

1. The distance must be positive definite.

2. The distance must be symmetric, so that the distance from x to y is the same as the distance from y to x. This is sometimes called the symmetry rule.

3. An object has a distance 0 from itself.

4. When considering three objects, x, y and z, the distance from x to z is always less than or equal to the sum of the distance from x to y and the distance from y to z. This is sometimes called the triangle rule.

Distance measures that obey the first three rules, but fail to obey rule 4 are referred to as semi-metric.

4.2 Clustering

When working with datasets containing tens and hundreds of thousands elements and even more, it can be hard to find any useful information without doing some sort of grouping in advance. This is what clustering theory is all about. The basic idea is to group similar elements together. The criteria by which the groups are formed are an essential part of the clustering techniques. For instance, if our experiment is an analysis of body-weight in a group of persons, it would probably be a waste of time doing cluster analysis with respect to the color of their clothes. However, if we want to see if there is any connection between body-weight and cholesterol level, we can divide the data into groups with respect to body-weight and "zoom in" on one group at the time. We could also make a bar diagram with the different weight-groups along the x-axis and the average cholesterol level in the groups as the height of the bars.

The methods presented in this chapter all belong to a group of methods known as unsupervised data analysis methods. Generally this means that analysis is done without any *a priori* knowledge of the input. Supervised methods on the other hand analyses the data with respect to known properties. Clustering algorithms exists for both unsupervised and supervised data analysis, but only those for unsupervised analysis are discussed here.

A problem with most of the clustering methods is that some input data are often forced into clusters even though they in reality do not share any similarities. Thus it can be important to analyze whether the data set exhibits a clustering tendency. Also, in some cases, the results of a cluster analysis need to be validated. However, these problems will not be discussed in this chapter. Readers interested in this field can for example consult the book "Algorithms for Clustering Data", by Jain and Dubes.

Figure 2. Clustering Example.

W = Within variance
B = Between variance
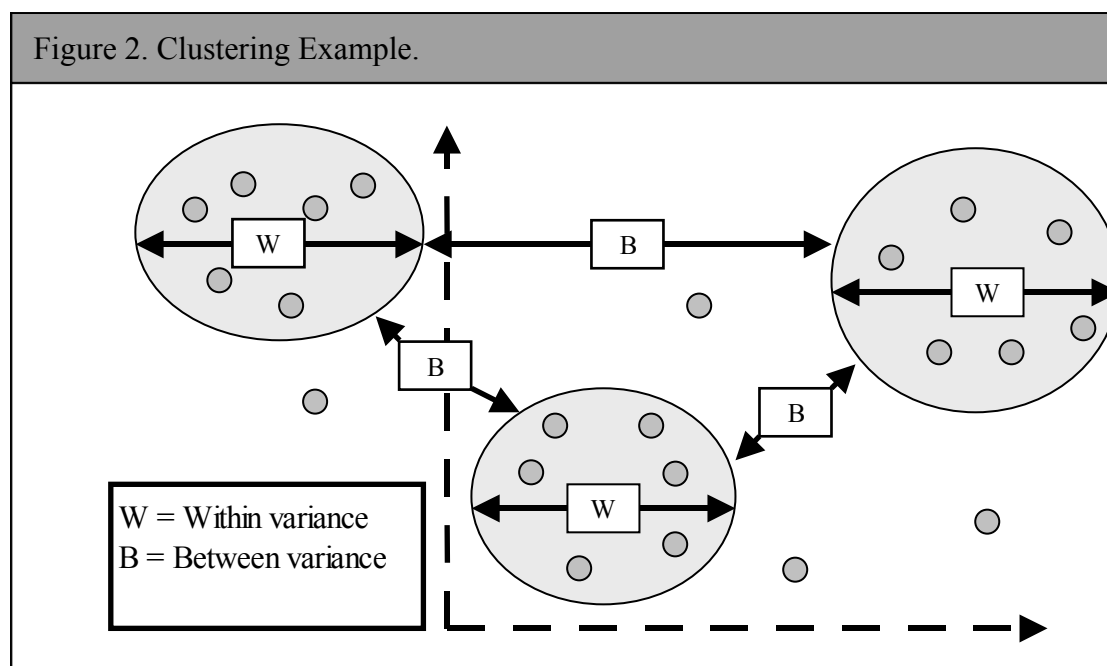
4.2.1 K-means clustering

K-Means is one of the simplest ways of doing cluster analysis. It simply creates k number of "boxes" (centroids) and assigns the input to them based on similarity. If two objects are relatively alike, they will be put in the same box, and the likelihood criteria will be updated according to the now new set of objects. The input can be in form of vectors, and each box is initially assigned one of these vectors randomly. In each step of the algorithm, all the input vectors are compared to the value of each of the boxes. The input vector that has the smallest distance to a box is then put in this box, and this box's centroid value (mean) is recalculated. A general k-means algorithm can be described in the following way:

1. Initially the input is arbitrarily divided into k centroids, and the reference vector (location) for each centroid is recalculated.

2. The input is rearranged so that each element is associated with the closest centroid according to some distance measure (e. g. **(5)**).

3. The new centroid location is recomputed for each subset.

4. Step 2 and 3 are repeated until no input point changes its association with a centroid, or an iteration threshold has been reached (the algorithm may not converge).

The k-means algorithm is very simple and fast, but it has some limitations. First of all, the number of clusters, k, must be given in advance. This can be a major disadvantage, because in some cases, this is exactly what we are looking for. To overcome this problem, the algorithm is often run multiple times with different k values, and the best results are kept. Another problem is the initialization of the algorithm where the different centroids are given a start value. As an iterative approach, the result of the algorithm is dependent on where the centroids are initially located. Some initialization approaches are listed below:

1. The Random approach: Divide the input into partitions of k clusters at random. This is the most used initialization method.

2. The Forgy approach: Choose k input at random as centroids and assign the rest of the input to the closest centroid.

3. The Macqueen approach: Choose k input at random as centroids and assign the rest of the input to the closest centroid, following the instance order. Recalculate the centroids after each assignment.

4. The Kaufman approach: Initial clustering is obtained by the successive selection of representative input until k initial centroids have been found. The first representative is the most central input point. The rest of the representatives are selected according to the heuristic rule of choosing the instances that promise to have around them a higher number of the rest of the instances, and have a relatively large distance from already chosen representatives.

4.2.2 Hierarchical clustering

There are generally two ways of performing this type of clustering, agglomerative and divisive. The divisive approach starts by defining the complete set as one cluster and

dividing it until each input element is the only member of a cluster. An agglomerative approach starts in the other end, with each input element as a cluster with a single member, and merges in each step two clusters until all are in the same cluster. J-Express Pro uses an agglomerative approach.

The result of a hierarchical clustering is normally a tree, also called a dendrogram. A dendrogram is a tree diagram displaying how the clusters are related. The leaves of the dendrogram is the input elements, and the root node is the final result of the clustering, containing all the leaves. A branch in this tree is a point where two clusters have been merged (or one cluster is split into two for divisive clustering). By cutting the dendrogram at a desired level, we get a set of disjoint groups (4.2.2).

Figure 3. Producing non-overlapping clusters by cutting the dendrogram.
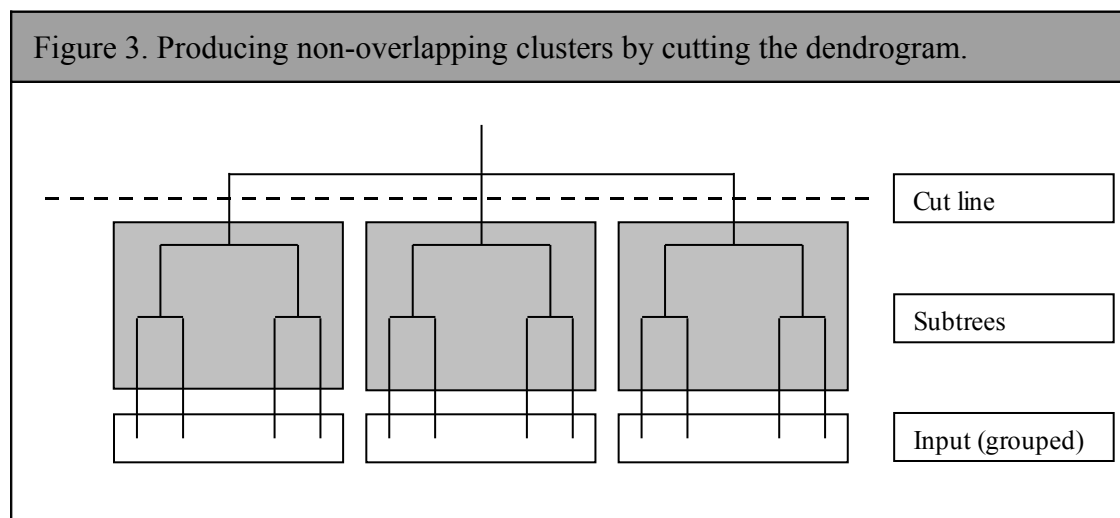
Hierarchical clustering is done according to some similarity measure (4.1). Initially, a distance matrix that contains a similarity score for all pairs of clusters (table 1 in 4.2.2) is created. If the input is in form of a set of vectors in a multidimensional space, this score is often based on the Euclidian distance **(5)** between two vectors containing values of similarities across multiple fields. By using a Euclidian distance (or another distance measure), a small value in the distance matrix implies that these two clusters/objects (defined by row and column number) are more similar than clusters/objects with greater value. When performing agglomerative clustering, the same matrix is scanned for the lowest value, which should be the smallest distance between two clusters. The cluster defined by the row (i) of the smallest element $d(i,j)$, and the one defined by the column (j) are then merged. The result of such merge is a new cluster containing both of the merged elements. The two merged clusters are then removed from the distance matrix and the new cluster is added. We now need some way of defining the distance from each of the elements already in the matrix to the new cluster. The three most used methods of doing this are called single linkage, complete linkage and average linkage (referred to in some contexts as nearest neighbor, furthest neighbor and centroid method respectively). Other methods can be defined by using different combinations of the distances involved in a clustering iteration.

Single linkage:

The distance between two objects is defined to be the smallest distance possible between them. If both objects are clusters, the distance between the two closest members are used.

This calculation is done by equation **(8)**. Single linkage often produces a very skewed hierarchy (called the chaining problem) and is therefore not very useful for summarizing data. However, outlying objects are easily identified by this method, as they will be the last to be merged.

Figure 4. Single linkage
Single linkage
$$d(C_{(ij)}, C_k) = \min\{d(C_i, C_k), d(C_j, C_k)\} \qquad \textbf{(8)}$$

The chaining problem:

In this example, new members are added to the cluster by the nearest neighbor function. We can see that the shape of the cluster is skewed.
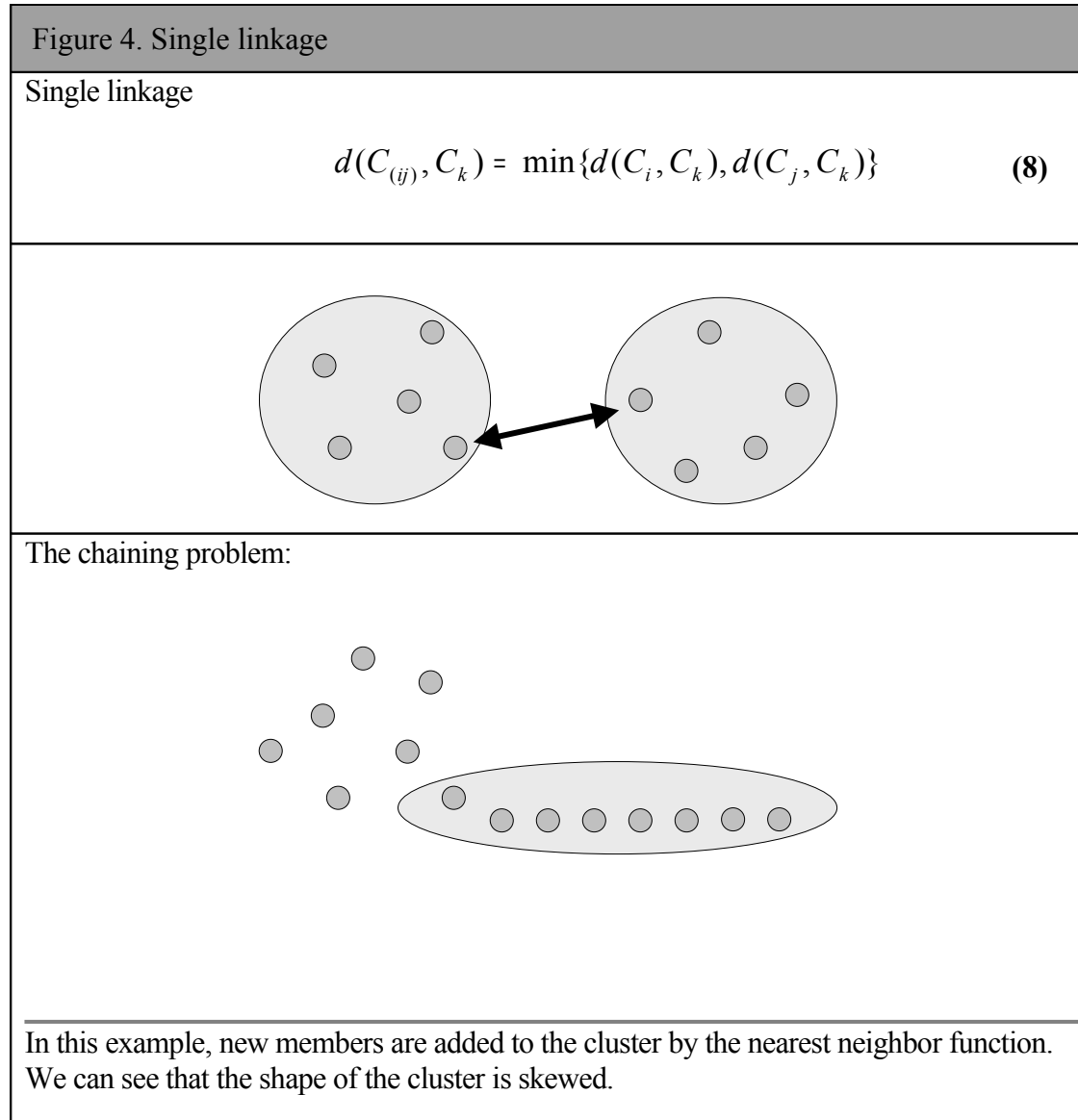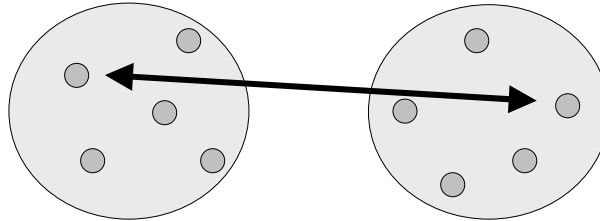
Complete linkage:

This method is much like the single linkage, but instead of using the minimum of the distances, we use the maximum. Complete linkage tends to be less desirable when there is a considerable amount of noise present in the data. Not surprisingly, complete linkage tends to produce very compact clusters.

Figure 5. Complete linkage
$$d(C_{(ij)}, C_k) = \max\{d(C_i, C_k), d(C_j, C_k)\} \qquad \textbf{(9)}$$

Average linkage:

This method takes the mean between all the objects in cluster i to all the objects in cluster j. There are several different ways of defining the average distance. In literature some of these are referred to as WPGMA (weighted pair group method with arithmetic mean), UPGMA (un-weighted pair group method with arithmetic mean), UPGMC (un-weighted pair group method centroid) and WPGMC (weighted pair group method centroid).
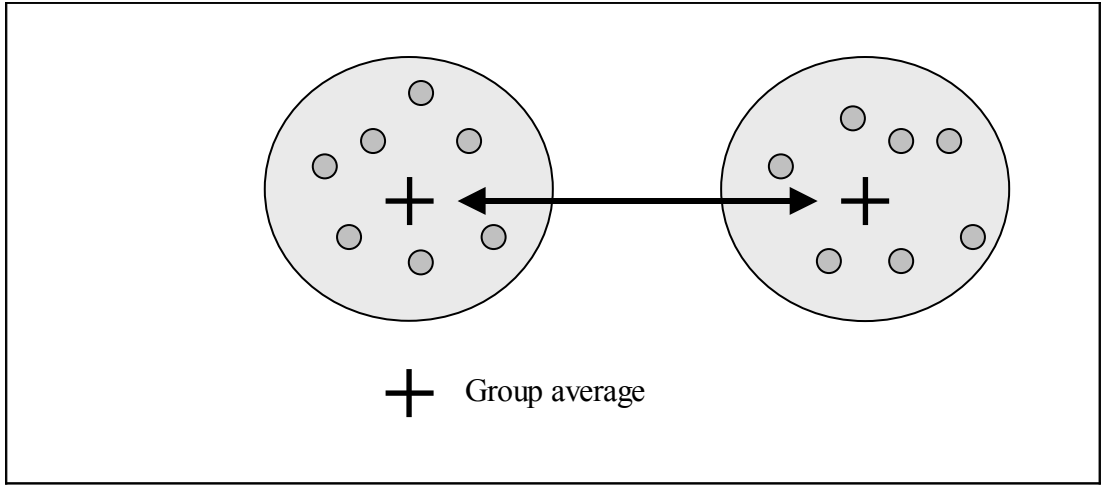
Figure 6. Average linkage
Average linkage (un-weighted average UPGMA)
$$d(C_{(ij)}, C_k) = \frac{d(c_i, c_k)|c_i| + d(c_j, c_k)|c_j|}{|c_i| + |c_j|} \qquad \textbf{(10)}$$
Alternatively, (weighted average WPGMA)
$$d(C_{(ij)}, C_k) = \frac{d(c_i, c_k) + d(c_j, c_k)}{2} \qquad \textbf{(11)}$$

Group average

A Hierarchical clustering Example

Figure 7. A simple hierarchical clustering example.

Table 1, *The first merge*

	Elm1	Elm2	Elm 3	Elm4
Elm 1	0			
Elm 2	6	0		
Elm 3	3	5	0	
Elm 4	4	4	2	0

Table 2, *The second merge*

	Elm1	Elm2	Elm (3 & 4)
Elm 1	0		
Elm 2	6	0	
Elm (3 & 4)	3	4	0

Table 3, *The third merge*

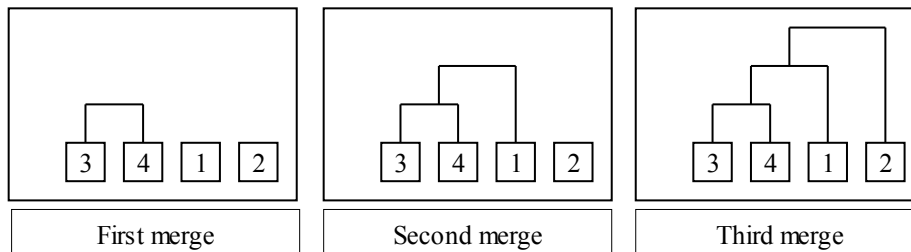|	Elm2	Elm ((3 & 4) & 1)
Elm 2	0	
Elm ((3 & 4) & 1)	4	0

Table 4, All elements are merged, clustering complete

	Elm (((3 & 4) & 1) & 2)
Elm (((3 & 4) & 1) & 2)	0

Dendrogram for this example.

This figure gives a small example of how to perform a hierarchical clustering with single linkage. There are four elements we want to cluster, and the numbers in the table is a measure of similarity (distance matrix). Note that we only use the lower triangular part of the matrix. (The details on the design and implementation of the distance matrix are further explained in their respective chapters). The initial data is as shown in table 1. In the first iteration, we search the matrix for the smallest element and find that this is the combination of element 3 and 4 (written in grey text). These two elements are merged in table 2, and because of our selection of linkage (single) the distances that are smallest to the other elements are kept. For example, the distance between our new cluster and element 1 is the smallest of the values 3 (Elm 1-Elm 3) and 4 (Elm 1-Elm4), which are 3. This operation is repeated in table 3, where

the merged element from the last iteration is merged with element 1. This procedure continues until all elements are merged into one cluster. When drawing the dendrogram, this final cluster will be the root of the tree.

4.3 Projection methods

Clustering methods reduces the amount of data items by grouping them. There exist also methods that can be used to reduce the dimensionality of the data, and present the data in a lower-dimensional system while preserving most of the variance. Examples of such methods are Multidimensional scaling MDA, Principal Component Analysis (PCA) and Factor Analysis. PCA will be further described below.

4.3.1 Principal Component Analysis (PCA)

The central concept in PCA is representation or summarization. In short, we want to reduce a set of variables to a set of linear functions that best summarize the original variables. However, there seems to be an infinitely number of linear functions that provide equally good summaries. In order to reach one unique solution, three conditions are introduced:

1. The derived linear functions must be mutually uncorrelated (orthogonal).

2. Any set of linear functions must include the functions of a smaller set (The best 4 functions must include the best 3 functions etc.).

3. The squared weights defining each linear function must sum to 1.

With these conditions, a set of *principal components* declining in importance can usually be found. By using all these components, a perfect representation of the data can be reconstructed. Using fewer will result in the best representation possible for that number of components. Each principal component is defined by an eigenvector (also called characteristic vector or latent vector) that defines this component as a linear combination of the original variables. Each eigenvector has a corresponding *eigenvalue* (*Definition 1*). If the original matrix is a correlation matrix, the eigenvalue of each component is its sum of squared correlations with the original variables. Each component's eigenvalue represents the amount of variance it will express.

PCA is also knows as eigen analysis. The data matrix is transformed into a set of vectors that span the same subspace as the original columns of the data. However, they are now characterized by a set of eigenvalues and eigenvectors. The transformation is done in form of a projection onto the selected eigenvectors.

Definition 1.

If A is an nxn matrix, then a nonzero vector x in the space R^n is called an eigenvector of A if Ax is a scalar multiple of x; that is,

$$Ax = \lambda x \qquad\qquad \textbf{(12)}$$

For some scalar λ. The scalar λ is called an **eigenvalue** of A, and **x** is said to be an eigenvector of A **corresponding** to λ.

The principal components for a matrix B are usually calculated from either the covariance matrix or the correlation matrix A (See EQN. 12). There is however no relationship between principal components obtained from a correlation matrix and those obtained from the corresponding covariance matrix. The covariance matrix of B is a matrix whose (i,j)th element is the known covariance between the i'th and the j'th element of the dataset. The correlation matrix is much like the covariance matrix, only with the correlation between the i'th and j'th element.

Algebra of an eigenvector projection

The eigenvectors of a covariance matrix R is a square n x n positive matrix. Its eigenvalues can be ordered as described above in the following way:

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0 \qquad\qquad (13)$$

The corresponding eigenvectors, defined as $c_1, c_2, \ldots, c_d$ are ordered accordingly. The m x d matrix of transformation is defined from the eigenvectors (principal components) of the covariance matrix as follows:

$$H_m = \begin{bmatrix} c_1^T \\ c_{21}^T \\ \vdots \\ c_m^T \end{bmatrix} \qquad\qquad (14)$$

The rows of H_m are eigenvectors. This matrix projects the original space into an m-dimensional subspace where the axes are in the direction of the largest eigenvalues as:

$$y_i = H_m x_i \quad \text{for i} = 1,\ldots,\text{n} \qquad\qquad (15)$$

The projected data can be written as:

$$C_m = \begin{bmatrix} y_1^T \\ y_2^T \\ \vdots \\ y_n^T \end{bmatrix} = \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_n^T \end{bmatrix} H_m^T = A H_m^T \qquad \textbf{(16)}$$

Where x_i is the original data, y_i is the corresponding projected data and A is the original data matrix.

The sum of the eigenvalues is the total variance in the original data, while the sum of the first m eigenvalues is the variance retained in the new space. Since the eigenvectors are ordered largest first, m could be chosen according to **(17)**.

$$r_m = \frac{\sum_{i=1}^{m} \lambda_i}{\sum_{i=1}^{d} \lambda_i} \geq 0.95 \qquad \textbf{(17)}$$

This will assure that 95% of the total variance is retained in the new space. Choosing the number of components to use can be difficult, and the equation above would in most cases resulted in a projection consisting of more dimensions than we are able to plot in a two or three-dimensional diagram.
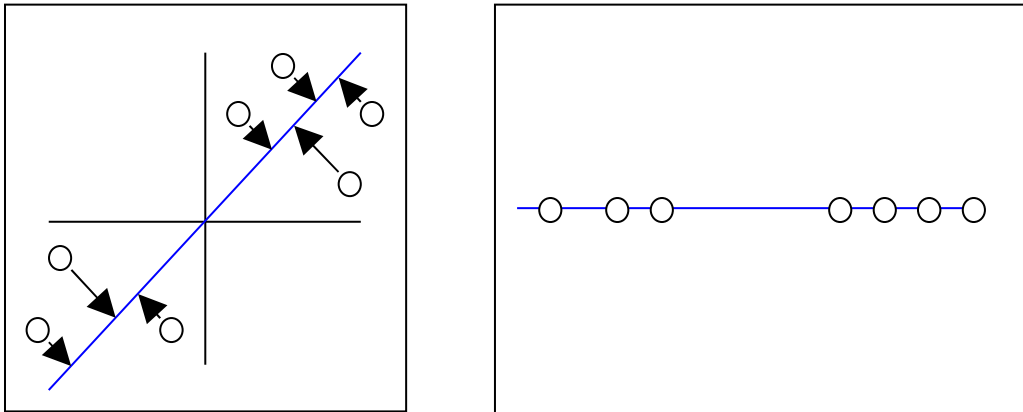
Number of principal components

If a set of d principal components is found for a data set, it is actually possible to use less that d components and still retain all the variance. This is however very rare. If we for any data set plot a curve with number of principal components used on the x-axis by the amount of variance explained on the y-axis, we usually get a plot like the one in. The principal components are usually sorted by their decreasing eigenvalues, and the components with the highest values are chosen. Again, plotting the amount of variance expressed by the eigenvectors sorted by their eigenvalue, will mostly result in a curve like the one in, although there are no direct relations between the two.

A PCA example

4.3.1 shows how a set of two-dimensional elements (circles) has been plotted in a two-dimensional coordinate system. The blue line represents the principal component for this set that expresses the most variance. When projecting the elements onto this component, we get the one-dimensional layout to the right in this figure. The arrows from the points to the principal component show how the projection is done.

Figure 8. A simple PCA example.

The objects (circles) in two dimensions (left) are projected onto the first principal component in blue (right).

4.4 Self-Organizing maps*

4.4.1 Principle

There are several different versions of the Kohonen Self-Organizing Map, but the principle is the same for all of them. Two different layers are described as the input layer and the neuron layer. The input layer is the data for which we want to find some pattern or groupings, and the neuron layer is a collection of neurons with relations both to other neurons in the layer and the data in the input layer. The idea behind SOMs is that the neuron layer through iteration steps called learning will adapt to the input layer in a way that reduces the complexity and makes it easier for humans to analyze.

The logical form of the neuron layer is often defined as a two dimensional grid. Each neuron has an x coordinate and a y coordinate relative to the other neurons in the net. This grid, called lattice, is usually a quadratic or a hexagonal net.

The neurons represent the inputs with reference vectors $\mathbf{m}_i$. One reference vector is associated with each neuron (i).

The SOM algorithm is based on iterations called learning. It starts by placing each neuron randomly in the input space by giving each neuron a reference vector equal to an arbitrary input. For each step in the learning process, a random input/value/point is selected and the best matching neuron (also called Best Matching Unit, BMU) from the neuron layer is found by calculating the distances from each neuron to the input value and return the one with the smallest distance **(18)**.

The best matching neuron *c* to input *x* is defined as (by Euclidian distance measure **(5)**):

$$c = c(x) = \arg\min_i \{\|x - m_i\|^2\} \qquad \textbf{(18)}$$

Where m_i is the vector of the i'th neuron in the neuron layer.

A "neighborhood kernel" (4.4.2) then decides how much each neuron (relative to the best matching neuron) will be moved in the direction of the input vector. Note that the algorithm actually operates on two different layers: a neuron layer and an input layer. The search for the closest neuron is done in the input layer. That is each neuron has a vector that is compared to the input (which has the same dimensionality). However, when deciding how much the closest neuron or its neighbors are to be moved, we calculate the distance from a neighbor to the closest neuron in the neuron layer. If the neuron network is defined as a two-dimensional lattice, this calculation is done in two dimensions.

This approach makes the lattice into an "elastic surface" that is stretched over the input, and only those neurons in the neuron layer that are topographically close to each other will learn from the same input.

---

* This section is mainly based on the theory behind Kohonen self-organizing feature maps.

The learning process:

$$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)] \qquad \textbf{(19)}$$

Where t denotes time, m_i is the vector of the i'th neuron, $x(t)$ is the input at time t and $h_{ci}(t)$ is the neighborhood kernel.

4.4.2 The neighborhood kernel

The neighbor kernel $h_{ci}(t)$ is a function defined over the neuron layer (the lattice points). It usually has the form $h_{ci} = h(\|r_c\text{-}r_i\|, t)$ where $r_c \in \Re^2$ and $r_i \in \Re^2$ are the radius vectors of nodes c and i, respectively, in the array. When $\|r_c\text{-}r_i\|$ increases, the magnitude of the function $h_{ci} \to 0$

The form of the neighborhood kernel can vary widely. Four possible versions will be described below. These are the bubble kernel, the Gaussian kernel, the cut-Gaussian kernel and the Epanechnikov kernel.

Bubble kernel

This is a very simple version of the neighbor kernel. It defines a width, or a radius from the best matching neuron and only those neurons in the reach of this radius is allowed to learn from the input. Another property of this version is that all the neurons within the radius are moved equally independent of their individual distances from the winning neuron. For a set of neurons Nc satisfying this criteria, we can write the function as:

The bubble kernel function.

$$h_{ci} = \alpha(t) \text{ if } i \in N_c \text{ and } h_{ci} = 0 \text{ if } i \notin N_c \qquad \textbf{(20)}$$

where $\alpha(t)$ is a decreasing function of time and N is the group of neurons that are close enough to learn from the input.

Gaussian kernel

The Gaussian kernel is the most used one, and it can be described as:

The Gaussian kernel function.

$$h_{ci} = \alpha(t) \times \exp\left(-\frac{\|r_c - r_i\|^2}{2\sigma^2(t)}\right) \qquad \textbf{(21)}$$

where $\alpha(t)$ and $\sigma(t)$ are decreasing functions of time, and σ defines the width of the kernel

An important property of this function is that the amount of movement towards the input will decrease as the distance to it increases. Thus, a neuron close to the input will be moved more than a neuron further away.

Cut-Gaussian

This is a combination of the two functions above. If the distance from a given neuron to the best matching one is within a given value (radius), it will be updated with the Gaussian kernel. If not, it will not be updated at all.

The cut-Gaussian kernel function.

$$h_{ci} = < Gaussian > \text{ if } i \in N_c \text{ and } h_{ci} = 0 \text{ if } i \notin N_c \tag{22}$$

where N is the group of neurons that are close enough to learn from the input.

Epanechnikov

This is a kernel that looks and works much like the Gaussian one, only the rate of movement decreases more as the distance from the input increases.

The Epanechnikov kernel function.

$$h_{ci} = \alpha(t) \times \left(1 - \frac{\|r_c - r_i\|^2}{\sigma^2(t)} \right) \tag{23}$$

where $\alpha(t)$ is a decreasing function of time, and σ is the width of the kernel.

4.4.3 The Elastic surface.

The form of the neighborhood function defines the stiffness on the elastic surface spanned by the neuron layer. Even if the neurons are initialized with random values, the form and "elasticity" of the neighborhood kernel will try to order the neurons in their respective locations in the lattice.

Figure 9. The form of the four neighborhood kernels described here.

| Bubble | Gaussian | Epanechnikov | Cut-Gaussian |
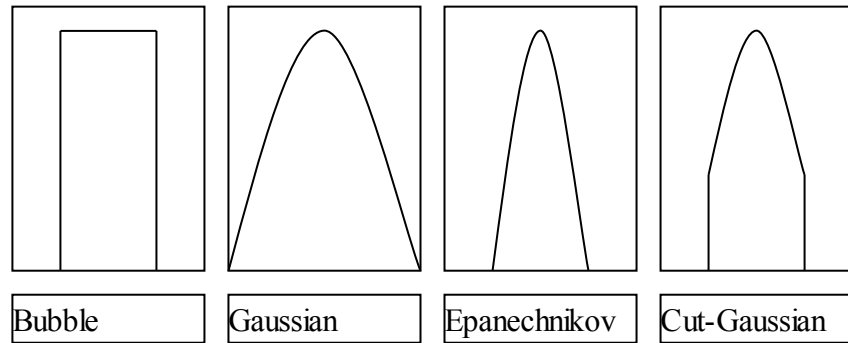
Figure 10. Examples of how the neighborhood functions work

Bubble   Gaussian   Epanechnikov   Cut-Gauss.
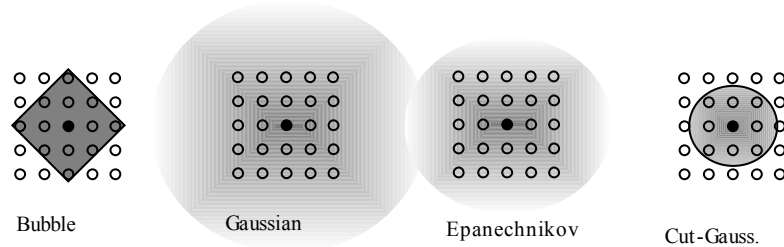
The dots are neurons in a square neuron net and the gray intensity corresponds to the amount of "pulling" towards the best matching neuron in solid black.

4.4.4 An example of the SOM algorithm

In this small example we shall see how the neurons are organized in a small self-organizing map of 3x3 neurons and a bubble kernel with width 1.
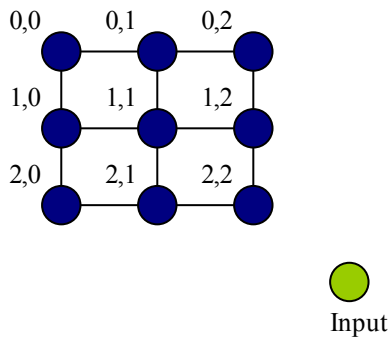
	The net has been initialized and run so that the neurons are ordered in the way they are to the left. Another input is read from the input layer and the algorithm finds the lower right neuron [2,2] to be the closest one. This neuron will be moved towards the input. The neighborhood function will now calculate the distances from every other node to node [2,2] in the lattice, and it finds that only two other neurons are close enough to be allowed to learn from this input.
The formula to the right shows the Euclidean distance calculation from neuron [1,1] to neuron [2,2]. The value returned is greater than 1.0, which is the limit in this step (using a bubble kernel). Therefore, this neuron will not be moved.	$$d(n[1,1], n[2,2]) = \sqrt{(2-1)^2 + (2-1)^2} \approx 1{,}4 > 1$$
	After this iteration, we can se that node 2,2 and its two closest neighbors are moved closer to the input. Note that this choice of neighborhood kernel will move all the three neurons the same distance.
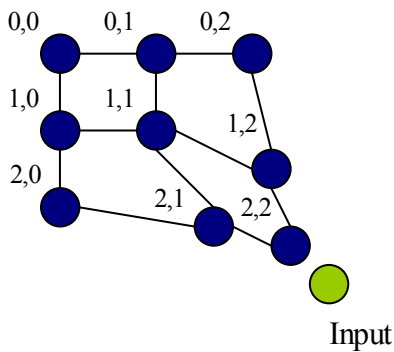
Figure 11. A SOM example.

5 Regular Expressions

Many of the search options in J-Express takes regular expressions as input. This enables the use of advanced search for more than one target at the time. In the beginning it may be a bit difficult to comprehend, but as you have used it a couple of times you will start to see the "endless" possibilities.

Regular expressions are created the following way:

5.1.1 Regular-expression constructs

Quick Examples

To find ID **YAL120W** use expression: **YAL120W**
To find all IDs **starting** with **YAL** use expression **YAL.***
To find all IDs **ending** with **W** use expression **.*W**
To find all IDs with **22 somewhere in the text** use expression: **.*22.***
To find all IDs with a **number somewhere in the text** use expression: **.*[/d]*.***

Usage

Construct	Mathces
Character	
x	The character *x*
\\	The backslash character
\0*n*	The character with octal value 0*n* (0 <= *n* <= 7)
\0*nn*	The character with octal value 0*nn* (0 <= *n* <= 7)
\0*mnn*	The character with octal value 0*mnn* (0 <= *m* <= 3, 0 <= *n* <= 7)
\x*hh*	The character with hexadecimal value 0x*hh*
\u*hhhh*	The character with hexadecimal value 0x*hhhh*
\t	The tab character ('\u0009')
\n	The newline (line feed) character ('\u000A')
\r	The carriage-return character ('\u000D')
\f	The form-feed character ('\u000C')
\a	The alert (bell) character ('\u0007')
\e	The escape character ('\u001B')
\c*x*	The control character corresponding to *x*
Character classes	
[abc]	a, b, or c (simple class)
[^abc]	Any character except a, b, or c (negation)
[a-zA-Z]	a through z or A through Z, inclusive (range)
[a-d[m-p]]	a through d, or m through p: [a-dm-p] (union)
[a-z&&[def]]	d, e, or f (intersection)
[a-z&&[^bc]]	a through z, except for b and c: [ad-z] (subtraction)

`[a-z&&[^m-p]]`	`a` through `z`, and not `m` through `p`: `[a-lq-z]` (subtraction)

Predefined character classes

`.`	Any character (may or may not match line terminators)
`\d`	A digit: `[0-9]`
`\D`	A non-digit: `[^0-9]`
`\s`	A whitespace character: `[ \t\n\x0B\f\r]`
`\S`	A non-whitespace character: `[^\s]`
`\w`	A word character: `[a-zA-Z_0-9]`
`\W`	A non-word character: `[^\w]`

POSIX character classes (US-ASCII only)

`\p{Lower}`	A lower-case alphabetic character: `[a-z]`
`\p{Upper}`	An upper-case alphabetic character: `[A-Z]`
`\p{ASCII}`	All ASCII: `[\x00-\x7F]`
`\p{Alpha}`	An alphabetic character: `[\p{Lower}\p{Upper}]`
`\p{Digit}`	A decimal digit: `[0-9]`
`\p{Alnum}`	An alphanumeric character: `[\p{Alpha}\p{Digit}]`
`\p{Punct}`	Punctuation: One of `!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~`
`\p{Graph}`	A visible character: `[\p{Alnum}\p{Punct}]`
`\p{Print}`	A printable character: `[\p{Graph}]`
`\p{Blank}`	A space or a tab: `[ \t]`
`\p{Cntrl}`	A control character: `[\x00-\x1F\x7F]`
`\p{XDigit}`	A hexadecimal digit: `[0-9a-fA-F]`
`\p{Space}`	A whitespace character: `[ \t\n\x0B\f\r]`

Classes for Unicode blocks and categories

`\p{InGreek}`	A character in the Greek block (simple block)
`\p{Lu}`	An uppercase letter (simple category)
`\p{Sc}`	A currency symbol
`\P{InGreek}`	Any character except one in the Greek block (negation)
`[\p{L}&&[^\p{Lu}]]`	Any letter except an uppercase letter (subtraction)

Boundary matchers

`^`	The beginning of a line
`$`	The end of a line
`\b`	A word boundary
`\B`	A non-word boundary
`\A`	The beginning of the input
`\G`	The end of the previous match
`\Z`	The end of the input but for the final terminator, if any
`\z`	The end of the input

Greedy quantifiers

$X?$	X, once or not at all
$X*$	X, zero or more times
$X+$	X, one or more times
$X\{n\}$	X, exactly n times
$X\{n,\}$	X, at least n times

X{*n*,*m*}	*X*, at least *n* but not more than *m* times

Reluctant quantifiers

X??	*X*, once or not at all
X*?	*X*, zero or more times
X+?	*X*, one or more times
X{*n*}?	*X*, exactly *n* times
X{*n*,}?	*X*, at least *n* times
X{*n*,*m*}?	*X*, at least *n* but not more than *m* times

Possessive quantifiers

X?+	*X*, once or not at all
X*+	*X*, zero or more times
X++	*X*, one or more times
X{*n*}+	*X*, exactly *n* times
X{*n*,}+	*X*, at least *n* times
X{*n*,*m*}+	*X*, at least *n* but not more than *m* times

Logical operators

XY	*X* followed by *Y*
X|*Y*	Either *X* or *Y*
(*X*)	X, as a capturing group

Back references

n	Whatever the *n*[th] capturing group matched

Quotation

\\	Nothing, but quotes the following character
\\Q	Nothing, but quotes all characters until \\E
\\E	Nothing, but ends quoting started by \\Q

Special constructs (non-capturing)

(?:*X*)	*X*, as a non-capturing group
(?idmsux-idmsux)	Nothing, but turns match flags on - off
(?idmsux-idmsux:*X*)	*X*, as a non-capturing group with the given flags on - off
(?=*X*)	*X*, via zero-width positive lookahead
(?!*X*)	*X*, via zero-width negative lookahead
(?<=*X*)	*X*, via zero-width positive lookbehind
(?<!*X*)	*X*, via zero-width negative lookbehind
(?>*X*)	*X*, as an independent, non-capturing group

Backslashes, escapes, and quoting

The backslash character ('\\') serves to introduce escaped constructs, as defined in the table above, as well as to quote characters that otherwise would be interpreted as unescaped constructs. Thus the expression \\\\ matches a single backslash and \\{ matches a left brace.

It is an error to use a backslash prior to any alphabetic character that does not denote an escaped construct; these are reserved for future extensions to the regular-

expression language. A backslash may be used prior to a non-alphabetic character regardless of whether that character is part of an unescaped construct.

Backslashes within string literals in Java source code are interpreted as required by the Java Language Specification as either Unicode escapes or other character escapes. It is therefore necessary to double backslashes in string literals that represent regular expressions to protect them from interpretation by the Java bytecode compiler. The string literal `"\b"`, for example, matches a single backspace character when interpreted as a regular expression, while `"\\b"` matches a word boundary. The string literal `"\(hello\)"` is illegal and leads to a compile-time error; in order to match the string `(hello)` the string literal `"\\(hello\\)"` must be used.

Character Classes

Character classes may appear within other character classes, and may be composed by the union operator (implicit) and the intersection operator (`&&`). The union operator denotes a class that contains every character that is in at least one of its operand classes. The intersection operator denotes a class that contains every character that is in both of its operand classes.

The precedence of character-class operators is as follows, from highest to lowest:

1	Literal escape	`\x`
2	Grouping	`[...]`
3	Range	`a-z`
4	Union	`[a-e][i-u]`
5	Intersection	`[a-z&&[aeiou]]`

Note that a different set of metacharacters are in effect inside a character class than outside a character class. For instance, the regular expression . loses its special meaning inside a character class, while the expression – becomes a range forming metacharacter.

Line terminators

A *line terminator* is a one- or two-character sequence that marks the end of a line of the input character sequence. The following are recognized as line terminators:

- A newline (line feed) character (`'\n'`),
- A carriage-return character followed immediately by a newline character (`"\r\n"`),
- A standalone carriage-return character (`'\r'`),
- A next-line character (`'\u0085'`),
- A line-separator character (`'\u2028'`), or
- A paragraph-separator character (`'\u2029`).

If `UNIX_LINES` mode is activated, then the only line terminators recognized are newline characters.

The regular expression `.` matches any character except a line terminator unless the `DOTALL` flag is specified.

By default, the regular expressions `^` and `$` ignore line terminators and only match at the beginning and the end, respectively, of the entire input sequence. If `MULTILINE` mode is activated then `^` matches at the beginning of input and after any line terminator except at the end of input. When in `MULTILINE` mode `$` matches just before a line terminator or the end of the input sequence.

Groups and capturing

Capturing groups are numbered by counting their opening parentheses from left to right. In the expression `((A)(B(C)))`, for example, there are four such groups:

1 `((A)(B(C)))`

2 `(A)`

3 `(B(C))`

4 `(C)`

Group zero always stands for the entire expression.

Capturing groups are so named because, during a match, each subsequence of the input sequence that matches such a group is saved. The captured subsequence may be used later in the expression, via a back reference, and may also be retrieved from the matcher once the match operation is complete.

The captured input associated with a group is always the subsequence that the group most recently matched. If a group is evaluated a second time because of quantification then its previously-captured value, if any, will be retained if the second evaluation fails. Matching the string `"aba"` against the expression `(a(b)?)+`, for example, leaves group two set to `"b"`. All captured input is discarded at the beginning of each match.

Groups beginning with `(?` are pure, *non-capturing* groups that do not capture text and do not count towards the group total.

Index