Genatomy – User Guide

Introduction

Genatomy is a visualization tool, aimed to meet the requirements of system biologists. It incorporates data from several sources, both experimental and biological, and allows the user a unified view which helps to find and understand the relations between data.

Currently, Genatomy does not contain algorithms to create regulatory programs, but it does contain views and tests to change and check results, as well as special functions and features that handle results in various formats.

Unlike other visualization tools, Genatomy "understands" biology, and does not handle data simply as Strings or numbers. It knows what a gene is and what the difference is between organisms. It connects with online databases and helps the user understand the biology behind the numbers.

Our vision is to create a tool which incorporates many of the functions that are usually run to test and change algorithm results and performed in computational programs, such as Matlab and R. From our experience, these changes are essential in order to understand and create better algorithms. Many of them are not used due to difficulties in connecting tools such as visualizer and algorithms with raw data, algorithm results and biological resources.

This guide describes most, if not all, of the functions and features of Genatomy, from the very basics of creating a new project to the specific functions used to handle results of various algorithms. If you are familiar with Genomica from Eran Segal's lab, you might find some of the explanations too basic, but fundamental differences between the softwares make even those parts essential for a good understanding of Genatomy. If you cannot wait to use Genatomy for your first project, you can skip some parts of the manual by following the yellow brick road, located at important parts of this guide. These paragraphs, marked in yellow, describe the basic concepts of Genatomy, and will help you create your first project step by step.

Table of Contents

Getting Started

Overview

First take a look at the main window:

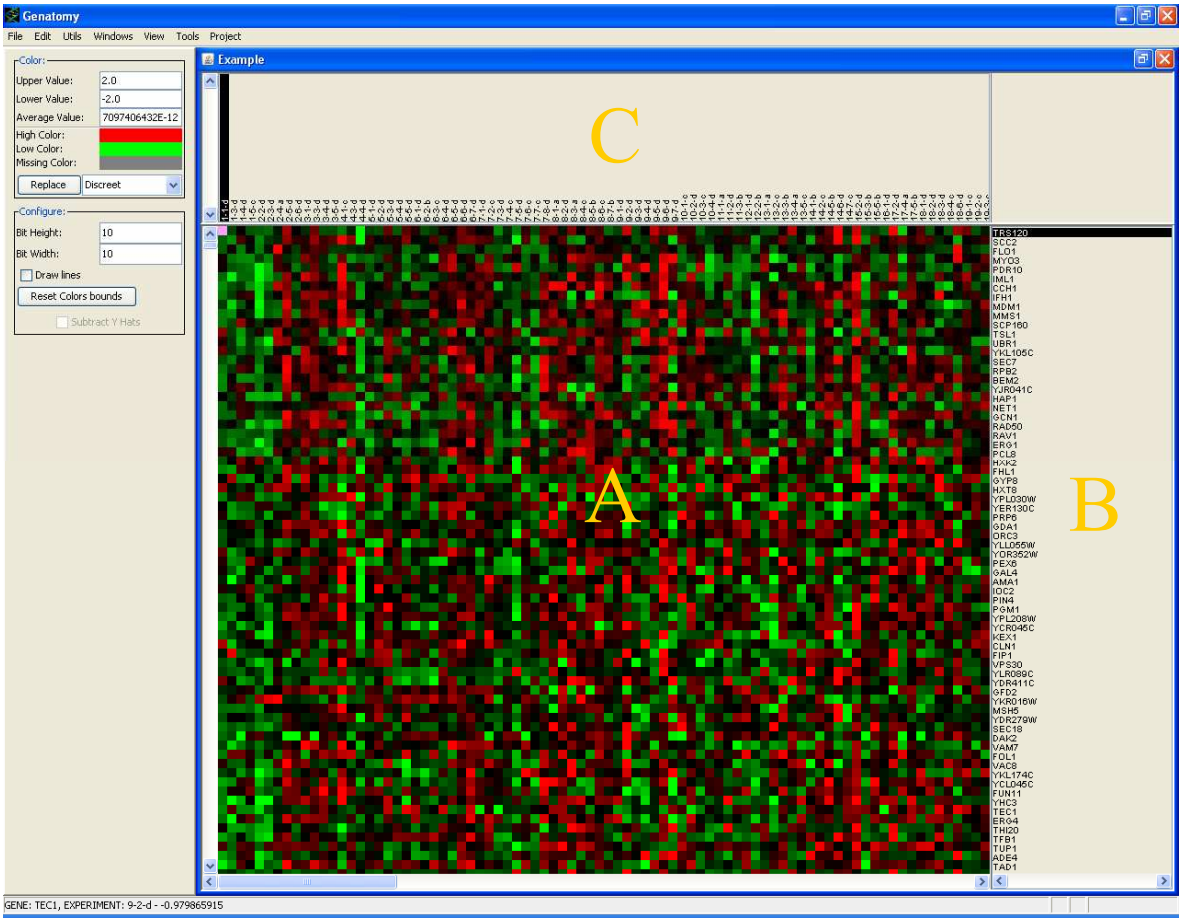


Fig. 1 – First look at a simple project

The viewer is divided into 2 main parts: the left part, called the side bar, provides information on selected features and enables the user to change the most common display properties. The right part, called the *main window*, contains the expression data window and other important windows such as enrichment results.

The project window, as shown in Figure 1 above is composed of the main panel which includes the expression data (A) and sub panels for other data objects, such as genes and sample names (B, C).

By clicking on each of the sub panels, different *Properties Panels* will be displayed on the side bar, such as colors and data options.

Running Genatomy

Genatomy is a java program, so it can run on virtually any operating system. It was tested on MacOS, Windows and Linux, with java versions 1.5 and 1.6. Genatomy is not compatible with older java releases, so please check your version by opening command line and typing "java -version". If you do not have java installed on your computer, or if you have a version older than 1.5, please go to http://java.sun.com/javase/downloads/index.jsp and install java.

Step 1: Downloading and Running Genatomy

Please go to http://www.c2b2.columbia.edu/danapeerlab/html/Genatomy/genatomy.html and download "Genatomy jar file". Save it and double click it.
If Genatomy does not open, you have a problem with your java installation.
You will also need to download the appropriate batch/shell file suitable for your OS. Save it in the same location as the jar file.
Double click the batch file and Genatomy will open again.
While you are there, you might want to download the example files (zip file) and unzip them in any directory. You will use these files to create your first project.

Although Genatomy can be run just by double clicking the jar file, this is not recommended. By default, java allocates 100MB of RAM to any java program, and that is not enough when using advanced features of Genatomy. The batch files run Genatomy with more RAM.

Creating your first project

Step 2: Creating your first project

Select from the menu File->New. A screen similar to Figure 2 will appear. To create the project, follow the next steps:

- Name the project.
- Choose an organism (the example files are for S. cerevisiae).
- Choose data type (Gene).
- Browse for the main data file.
    - You already downloaded the file in step 1. Go to the directory where you saved it and choose mainfile.tab.
- Click "Next" twice and then finish.

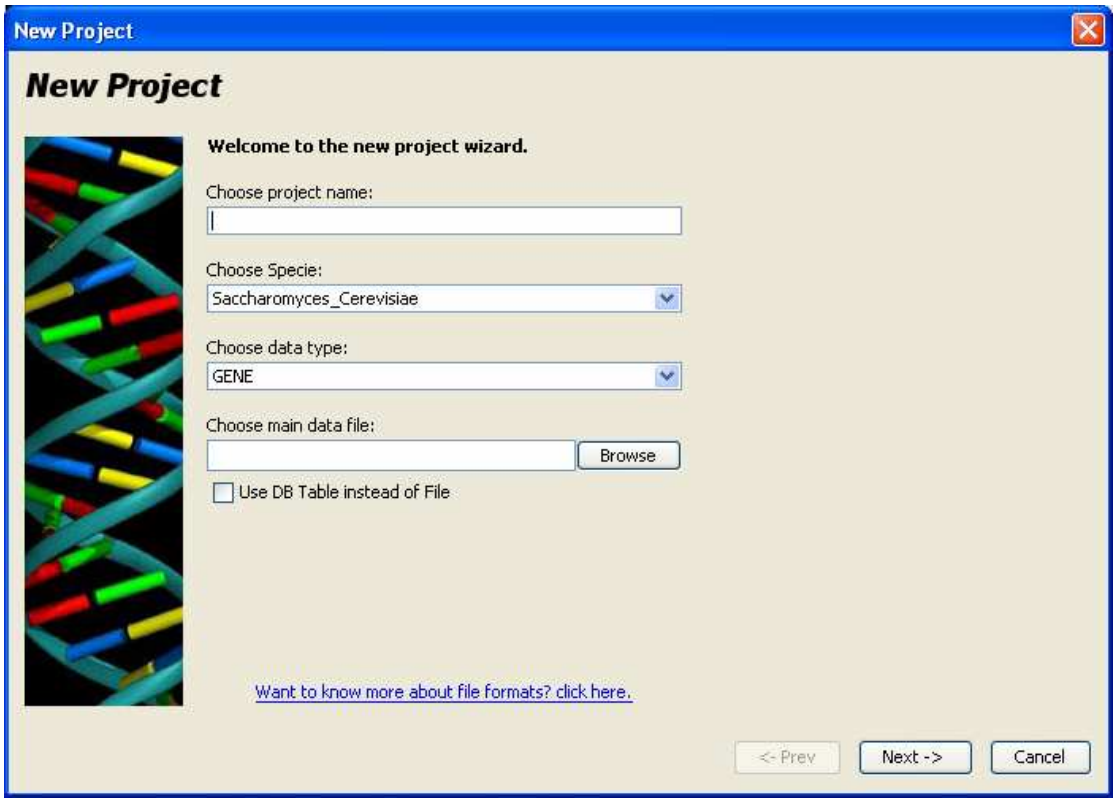


Fig. 2 New Project Form

A window similar to Figure 1 will open. You can select data points and see their values in the status bar at the bottom of the window.

You might notice that the gene list does not contain any information about the genes, and the names are symbol names for S. cerevisiea. You will have the opportunity to change that later.

Step 3: Save your project

- Select from the menu File->Save as.
- Name the project file and save it.
- The project now appears in the menu "File->Recent projects".

Notice that the saved project file (extension ".gpf") does NOT contain the expression data and only has links to the files loaded. The file is an XML formatted file and can be viewed or edited with any XML viewer.

Controlling colors and other display properties

Each display panel (or section) has its own display properties and a different *Properties Panel*. The properties panel of the main display panel (Fig 1.A) is displayed in Figure 3.

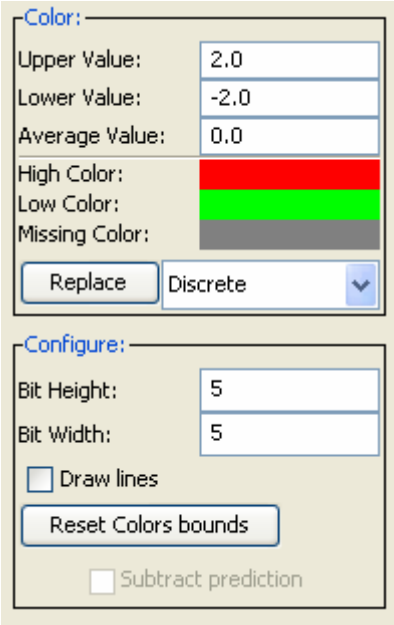


Figure 3 - Main view Properties panel

The lower panel controls the display features of the data panel, while the upper one controls the colors. The *color panel* looks different for every coloring scheme (Read more here). The panel in Figure 3 is used for gradient coloring.

In gradient coloring, the average value is always black. Values between the average value (probably 0) and the highest value (2 in that case) get a different shade of the chosen color (red in Figure 3). Likewise, all values lower than the average get a shade of green. Values lower than the lowest get the selected "Low color" (green).

We will now review all the properties and options of these two panels:

- *Colors panel*:
  - Upper, Lower and Average values – the values for the highest, lowest and average colors.
  - High and Low Colors – the colors for the highest and lowest values.
  - Missing Colors – The color for missing values in the data.
  - Replace button – Replaces the coloring scheme. Read more here.
- *Configuration Panel*:
  - Bit height and width – Dimensions of the squares in pixels. Also controls the font size in panels B and C. (Figure 1).
  - Draw lines – Draw separation lines between squares.
  - Reset color bounds – Setting the upper, lower and average values according to the values of the data displayed.

Step 4: Changing display properties

Select the main panel (Fig 1.A). Genatomy figured that the data loaded is continuous data, so it selected the gradient coloring method.
Change the upper and lower values and press enter. See how the display changes.
Now change one of the values and press CTRL+ENTER, both values are changed.

Genome Information

How does it work?

Genatomy can load and use information about genes, including unique gene ID, symbol name and genome location. Moreover, Genatomy handles genome data from different organisms differently. For example, right clicking on a gene name will give you specific options such as opening the gene's data page in the main website of that organism.

Genatomy can load that information from different files types:

- Conversion table – File containing gene name and aliases.
- Description table – Description of genes.
- SNPs data file.
- Full Genome Information - one file containing all genome information. Format specific to each organism. In most cases it is downloaded from the organism's web site (such as sgd_features file from the SGD web site).

In most cases, loading the *full genome information* file is the best option. The number of data files is not limited so you can always load the full data file and add additional data using other files.

When creating a new project (or when importing GXP file), you will have the option to choose information files. When changing data files after creating the project, you will need to close the project and reopen it in order for the changes to take effect (Genatomy suggests to do so automatically).

Genatomy matches gene names using these tables. For example, if you load an expression file which uses ORF as row (genes) names, and then you load an annotation file which uses gene symbols, Genatomy will match the gene symbols to ORFs if given the appropriate conversion table. See Figure 4.

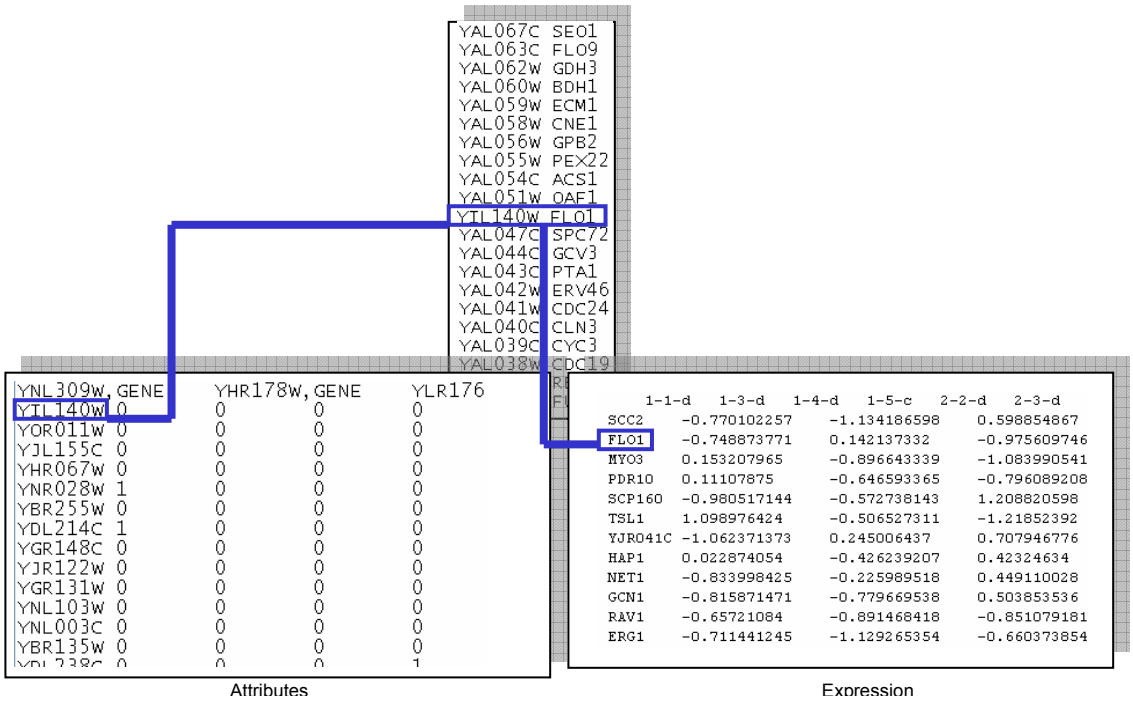


Figure 4 – A demonstration of two data files, each one uses different gene names, and a conversion table helping to match the data.

Repository and Local files

Dana Pe'er's lab manages a file repository containing *full genome information* files for several organisms. Whenever you are connected to the internet, you will be able to download those files to your computer and use them.

In addition, if you need to work with certain information files often, you can add them to your local repository and load them automatically to any project you create (see below).

In the second page of the *New Project wizard*, a list of local and online files will be presented (Figure 5). Here you can select which data files to load from the repository, and you can load additional files specific for the current project. You can always access this window from the menu Project -> Data files.

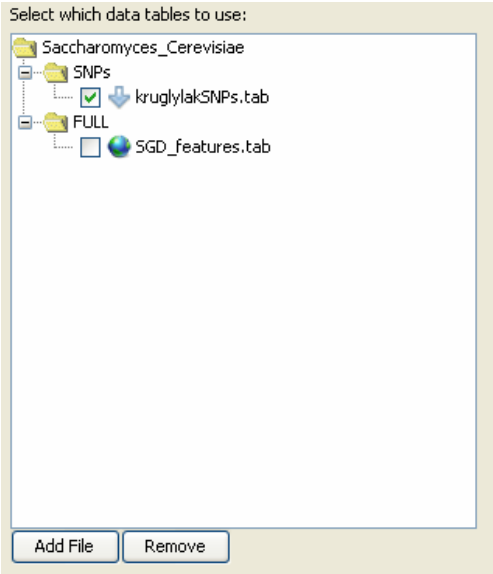


Figure 5 - A tree for selecting data tables. Files that are available only online will be presented by a globe, and files with newer version online will be marked with an arrow

Step 5: Adding data on genes

Go back to your project and click on "Project -> Data tables".
Since we are working with S. cerevisiae, you now see data files available for that organism in our file repository.
We will now add the full list of genome data by selecting SGD_features.tab and OK.
You will now be asked to download the file to your computer, so (please) approve the download.
As mentioned above, after making any changes to the project's data files, you need to reload the project. Genatomy will do that for you, just click OK.

Now you can see that the genes list contains data about each gene, including location, unique identifier, description and aliases.

If you want to manage your local repository (only for manually added files), go to File -> File repository.

If you choose files that are not on your local computer, or if you load a project that uses files with a newer version online, you will be asked if you want to download them.

Genatomy also makes it easier to share project files. If you are using files from the web repository, you do not need to send those files with your project. Genatomy will download them automatically.

Other Data types

Up until now, we worked with expression files only, in which each row represents a gene. Genatomy can load other data types, such as growth data, annotations, genotypes (gene markers) and more. A project can contain several data types, and each one will be treated differently by Genatomy. The basic feature for all of the above is coloring – each data type in the project will get its own coloring scheme; for example, you will see expression as red and green, attributes as blue and white, and growth curves as yellow and purple.

Several data types get special care:

- Genes of course are treated as genes, including genome location, unique ID, description, etc.
- Gene markers are treated differently as well and MUST be named using a certain convention – MX_Y_Z, where X is the chromosome ID, Y is the starting point (in bps), and Z is the last base pair. For example M1_2_3000 represents a marker present on chromosome 1, between base pairs 2 and 3000. A special feature for gene markers is the tooltip, which contains a list of all genes in that area. Right clicking on a gene marker will open a window with that list, and an option to change the margins of the markers to include genes further away from the marker. The number specified in this window is global and will affect all gene markers in the project (Figure 9).

In order to specify the type of data, write a comma and the type (from Table 1) after the feature name. For example: ORF113W, GENE. You may recall that when creating a new project, you need to specify the default type. The selected type is the default type for features in rows of data files, EXPERIMENT is always the default for columns, and ATTRIBUTE is the default for columns of attribute tables. Read more about file formats here.

Available data types
GENE
EXPERIMENT
PHENOTYPE
ATTRIBUTE
GROWTH_CURVE
GENE_MARKER
OTHER

Table 1. Available data types

Attributes

Overview

Attributes are any data added to the features loaded in the main file. The most commonly used source of attributes is Gene Ontology (GO), in which each gene is annotated with its functions in the cell.

To avoid ambiguity, we are using the term *attribute* and not annotation.

Genatomy can load and manage attributes both for the rows (genes most of the time) and for the columns (samples or experiments). It can load more than one table of attributes, and load non-binary attributes. For example, if you have data about a certain SNP in all samples, you can load it as a four-value attribute (A,C,G,T) and not just 0 and 1. You can also load continuous data, like growth curves, or even gene expression as attributes.

In Figure 6 you can see the project from figure 1, this time with gene (D) and sample (E) attributes.

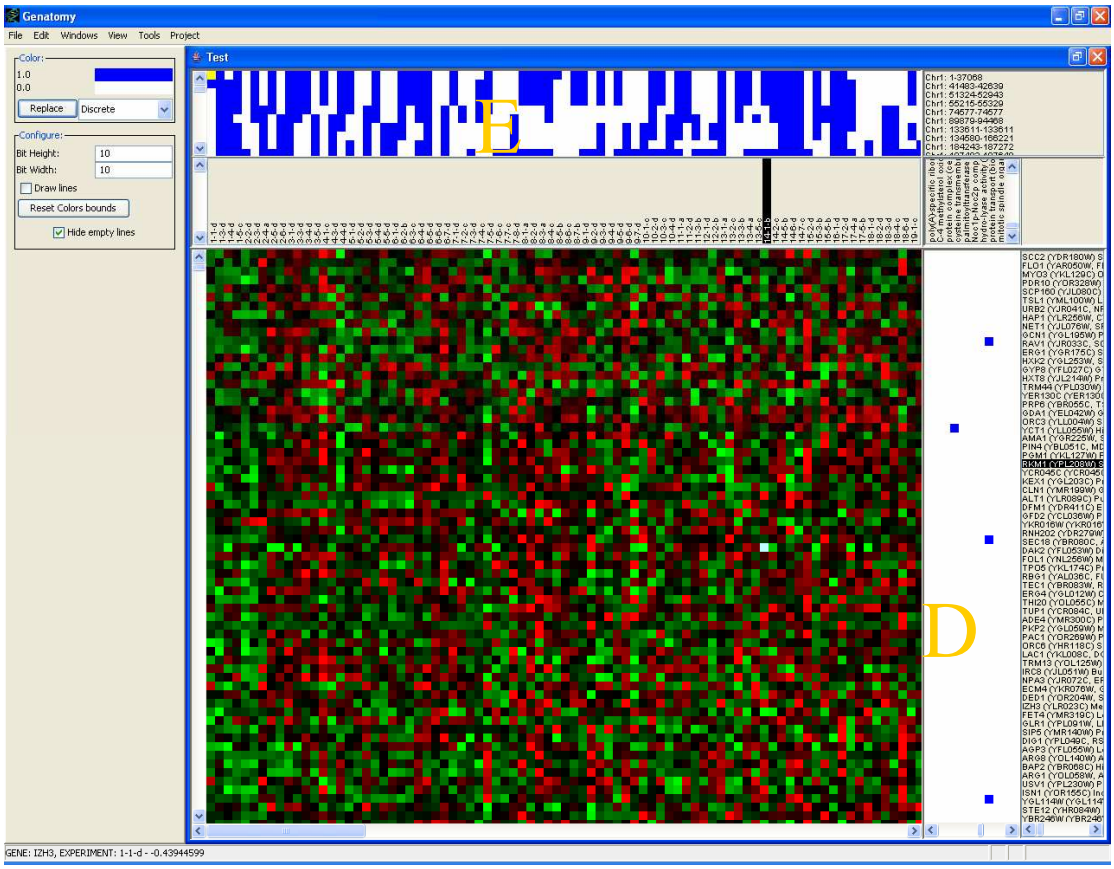


Figure 6 – Display with genes and samples attributes.

Project Attributes

The window (Figure 7) that handles the project's attributes can be accessed through "Project->Attribute Manager".

The file repository also contains attributes files for many organisms. As in data files, the repository files are represented by a globe icon or an arrow when a newer version is available. (See figure 7).

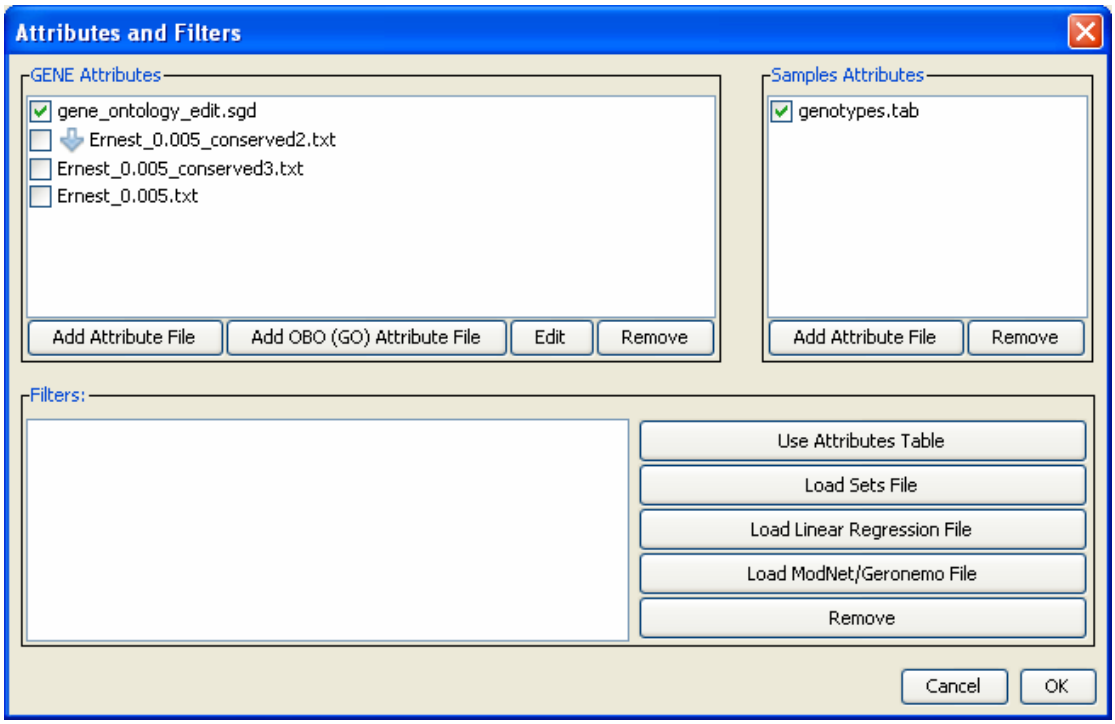


Figure 7 – Attributes manager form. The upper section manages attributes (genes or rows on the left side, samples or columns on the right side). The lower part manages filters.

Step 6: Adding Gene Attributes

Add project GO annotations from the repository by going to Project-> Attributes Manager.
You will see several files from the repository, all with a globe icon, which means that you do not have these files on your local computer.
Choose gene_ontology.sgd and press OK. The manager will ask you to download the file.
Go to the menu View and click on Gene Attributes to see the attributes.
Save the project.

Now that you added gene attributes, you will get a window similar to Figure 6. Click anywhere on the gene attributes panel (Fig 6.D) and review the properties on the side bar (Figure 8).

Notice that the *Color Properties Panel* is now different. This is the panel for discrete data in which each of the values gets a different color.

The *Configuration Panel* is very similar to the one in the *Main display* (Figure 3) except for the "Hide empty lines" check box. If there are attributes with no affiliated features in the current view, checking the box will remove them from view. This is useful when looking at a smaller group of genes (like a specific module that may not have any genes in an unrelated attribute).

When adding Gene Ontology file from your computer ("Add OBO (GO) Attribute file" button), you will be asked to choose one file only, although the annotations come in two files, one describes the ontology (obo file) and the other describes the annotations of the genes. You need to select the second file, and you have to make sure the obo file has the exact same name (except for the suffix), and can be found in the same directory.

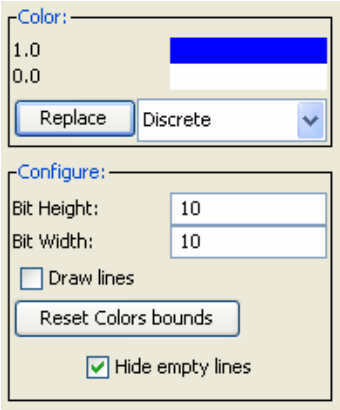


Figure 8 - Attributes Properties Panel

Step 7: Adding Samples Attributes

You will now add sample attributes to your project, in this case, gene markers.
Return to the attributes manager form and click on "Add Attribute File" under the Samples Attributes section.
Browse for the file "genotypes.tab" from the examples zip file.
Press OK.
Now add the samples attribute view under View->Samples Attributes.

Gene-marker attributes get special care in Genatomy. The attribute is converted into genome location, and Genatomy can now use this data to show you the genes in or around that marker.

Move your mouse cursor over the *gene-marker* names and wait for the tooltip to appear. The tooltip contains all genes in that marker area (over the marker, inside it or partially inside it).

Right click on one of the markers and choose "Open Data Window". A window containing the list will appear (Figure 9). In this window you can change the margins of the marker, so genes around the marker (+/- the number written in the box) will also be included. This number is a global number and is shared by all the markers.

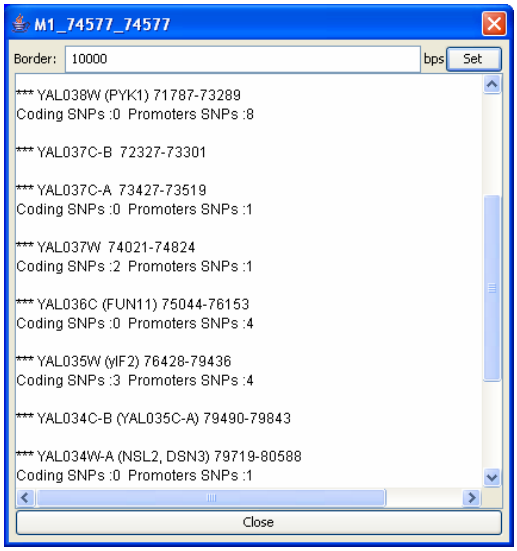


Figure 9 – Genes around/in/partially in a certain gene marker

Filters

"Filters" is a way to filter out data and view only part of it (both on the gene and samples axis). Each *filter* contains *modules*, where each *module* describes a set of genes and samples that you want to display.
A module can also contain regulatory information (such as module network or linear regression), and also specifies the order of genes and samples.
Once the project has filters attached to it, another panel will appear on the side bar (Figure 10).

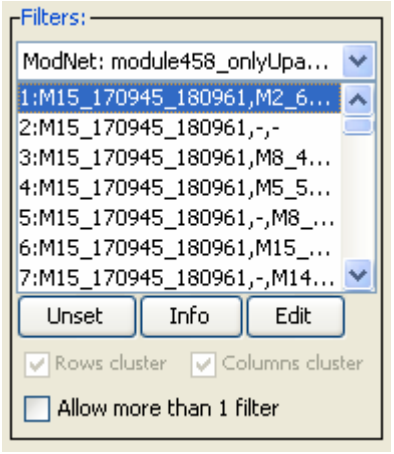


Figure 10 – Filters panel

We will now review the features of the panel:

- The combo box lists all filters.
- The list box contains all modules of the selected filter.
- Selecting a module applies it to the view.
- "Unset" unsets the module and cancels the filtering on the data.
- Edit allows you to change the name of the module (or give it a nickname), and to add notes on the module (which are displayed as a tooltip).

By default, only one module can be selected and filter the data. If, for some reason, you want to apply two filters, check "Allow more than 1 filter" box (highly NOT recommended). Use this feature only when you select two different modules to filter samples and genes.

By Attributes Filter

The simplest type of filters creates modules from an attributes table, where each module is an attribute, and contains all genes (or samples) with a value greater than 0 for that attribute (which means that it is annotated in binary data).

Step 8: Creating "By Attribute Filter"

Go back to the attributes manager window (Figure 7), select the "gene_ontology" attribute table and click on "Use Attributes Table".
A new filter will appear in the filters list.
Click OK and a *filter panel* will appear on the side bar (Figure 10).
Save the project.

Module Networks

Another type of filter is the *Module Network filter* (Segal, Shapira et al. 2003). A module network module contains a set of genes (or other features) and samples, governed by a regulation tree.

Each branch of the tree divides the samples to two groups, using a rule on a regulator, such as, all samples where the value of a specific gene is lower than 0, go left, and the others go right.

Step 9: Loading Module Networks filter

Go back to the *attributes manager window* (Figure 7), and click on "Load Modnet/Geronemo file".
Choose modnet.xml from the examples directory.
Click on OK and choose modnet.xml form the combo box in the *filters panel* (Figure 10).
Go to "View" menu and choose "modnet view"

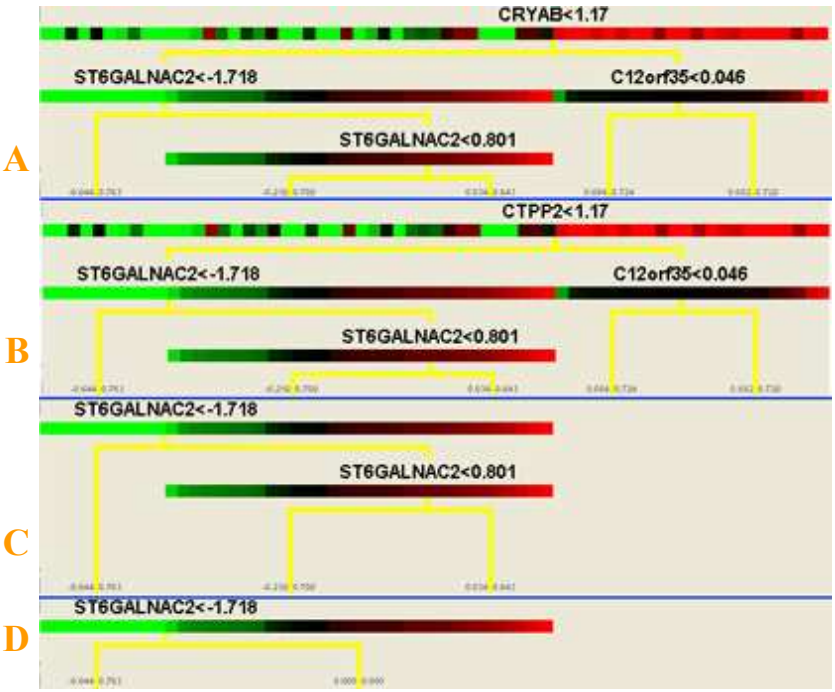


Figure 11 – Module networks panel. A – Full tree, B – same tree with an alias for the first regulator, C – part of the tree (zoom in), D – The branch from C without sub-branches.

Genatomy allows loading *module networks files* (an XML file) and displaying them. It allows users to perform a few changes to the regulatory trees:

- Zoom in and display only part of the tree (Double click on a branch).
- Display other names of a regulatory gene, if it has aliases (Left click a regulator).
- Remove part of the tree (Right click on a branch).

Linear Regression (and non-linear interactions)

Another type of modules is *linear regression modules*. These are sets of genes (or features) and samples, governed by a set of regulators creating a linear dependency between them and the features.

As in every filter file, this file is also in XML format, and the modules can be created in any program outside Genatomy.

Figure 12 shows what the *Linear Regression panel* looks like (A), and what happens to the *filters panel* (B) when a Linear Regression module is chosen.

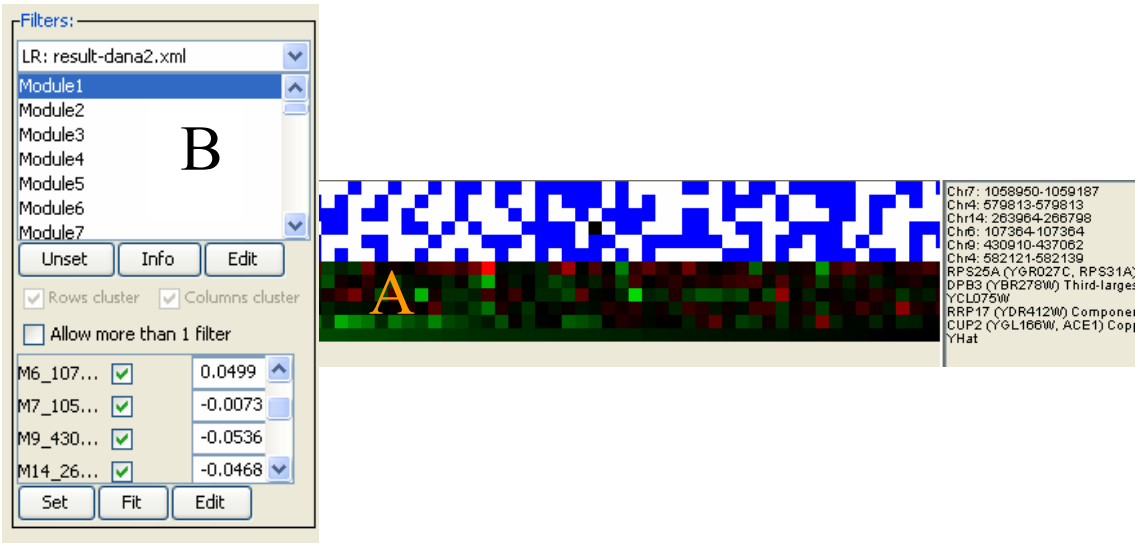


Figure 12 – A – Linear regression data panel, B – Filters panel with Linear Regression module.

Step 10: Loading Linear Regression filter

Go back to the *Attributes Manager window* (Figure 7), and click on "Load Linear Regression file".
Choose lr.xml from the examples directory.
Click OK and choose lr.xml form the combo box in the *filters panel* (Figure 10).
Go to "View" menu and add "LR panel"

Genatomy calculated the prediction (Y hats) using coefficients, and orders the samples from the smallest to largest value of the prediction.

As shown in Figure 12, the *filters panel* displays a list of regulators and their coefficients under the modules list. The coefficients can be loaded to Genatomy or placed as input by the user. By clicking the *Set* button, the prediction is re-calculated.

The user can also choose to exclude regulators by deselecting the check box next to the regulator name. For the changes to take effect, the user must use the *Set* button.

Genatomy can also perform linear fitting. By clicking *Fit*, Genatomy calculates linear least squares fitting between all the checked regulators and each one of the features in the module. The average of the coefficients is taken and the prediction is recalculated.

Looking back at Figure 3 we can see a disabled check box labeled with "Subtract prediction". The check box becomes enabled when selecting a *linear regression module*, and by choosing it the values of the prediction are subtracted from the values of the module features, helping us to see how good the fitting is (closer to 0 means better).

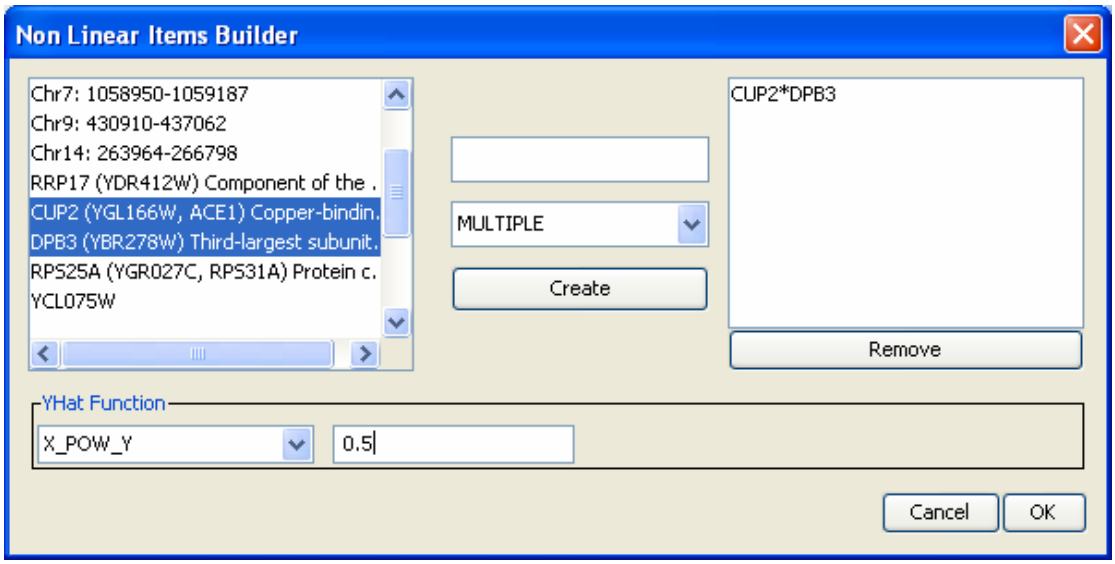


Figure 13 – Non-linear interactions window

Genatomy also allows you to add interactions between regulators as new regulators. Choosing *Edit* in the linear regression part of the *filters panel* opens the window shown in Figure 13.

For example, you can choose 2 regulators, choose any interaction such as multiply, divide, etc. and create a new regulator, where the data is the result of the interaction chosen.

An important feature for all modules with regulatory information containing *gene-markers* is the genomic information added to genes in the module.

Each gene marker gets a color, and all genes around that marker are marked by squares of that color. The shade of the square is correlated with the gene distance from the marker. Placing the mouse over a square displays the exact distance from the marker at the bottom of the window.

Simple Sets

Another type of filter is the *simple sets filter*, in which the module is only a set of features and samples, without a regulatory program. A *simple set module* allows you to perform actions on both axes (features and samples) such as clustering.

Since the *module networks file* and the *linear regression file* both contain a basic module file structure, they can both be loaded as *simple sets filter*.

Clustering

Genatomy can perform a hierarchical clustering on both axes, using several functions. The clustering trees are shown in special panels.

Step 11: Clustering one module

Choose any one of the modules loaded in steps 9 or 10. Make sure that your module contains more than 1 gene.
Click on the menu item "Project->Cluster". A window similar to Figure 14 will appear.
Choose "Cluster current module" and click "Run".
Wait a few seconds. The results panel will appear on the right and the genes will be displayed in a different order.

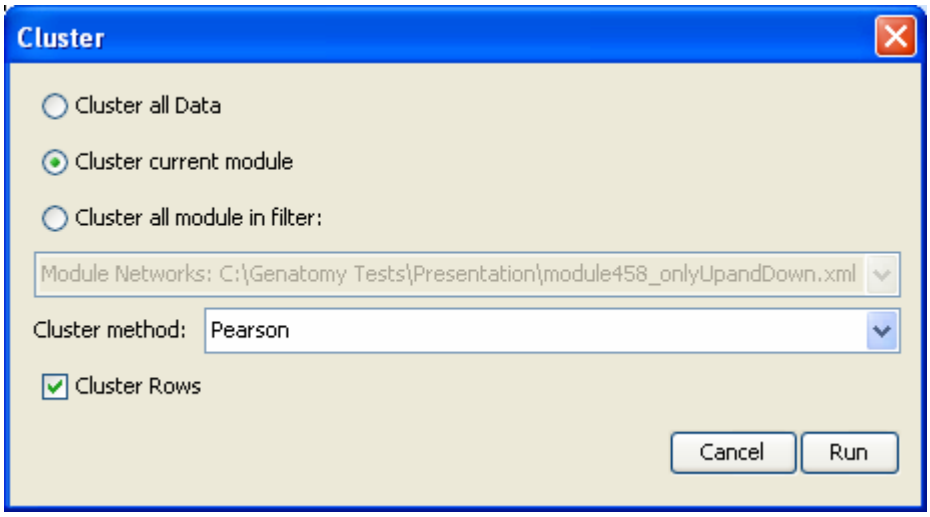


Figure 14 – Clustering form

In Figure 14 you can see the options of the clustering function. You can perform clustering on all the data from the main data file, only on the selected module, or on all modules from a specific filter file.

You can also choose the method to use for the clustering – currently available functions are: Pearson, Absolute Pearson and Euclidean distance.

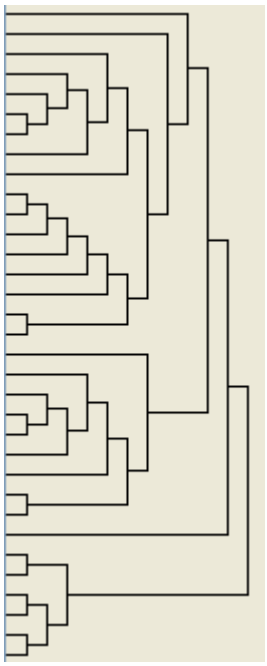
To perform clustering on columns, uncheck the "Cluster Rows" box.

Notice – performing clustering on columns is only available for modules without regulatory information, so you cannot cluster columns of *module networks* or *linear regression modules*.

Figure 15 shows the *clustering results panel*. By moving the mouse over the results tree, information about the tree appears on the bottom of the window. You can also zoom in on the tree and view only a subset of rows and columns.

The *Configuration Panel* of the *clustering results panel* also contains an option, called "solid color", and by deselecting it, the branches of the tree are colored by the value of the clustering function for that branch.

The clustering results are saved and loaded with the project.



L.L.R.L.L.L.R.L.R.R.L.R.R.L.L.L.L.R.L.L.R.L.R.L.L.L.L.R.R.R.L.L.L.L.L, Score: 0.7539058915333449, Level: 1

Figure 15 – Clustering results. A. Clustering tree. B. Information about a branch in the tree including path, function score (in this case Pearson) and the level of the branch.

Filters and Attributes options

Hyper Geometric Enrichment

Genatomy can calculate *hypergeometric enrichment* in both axes. There are two ways of viewing the p-values. The first is a table with p-values for all modules vs. all attributes (gene sets), as shown in Figure 18. The other is *on-the-fly* p-values, which are calculated after every change to modules and presented as red squares next to the attribute names, as shown in Figure 20.

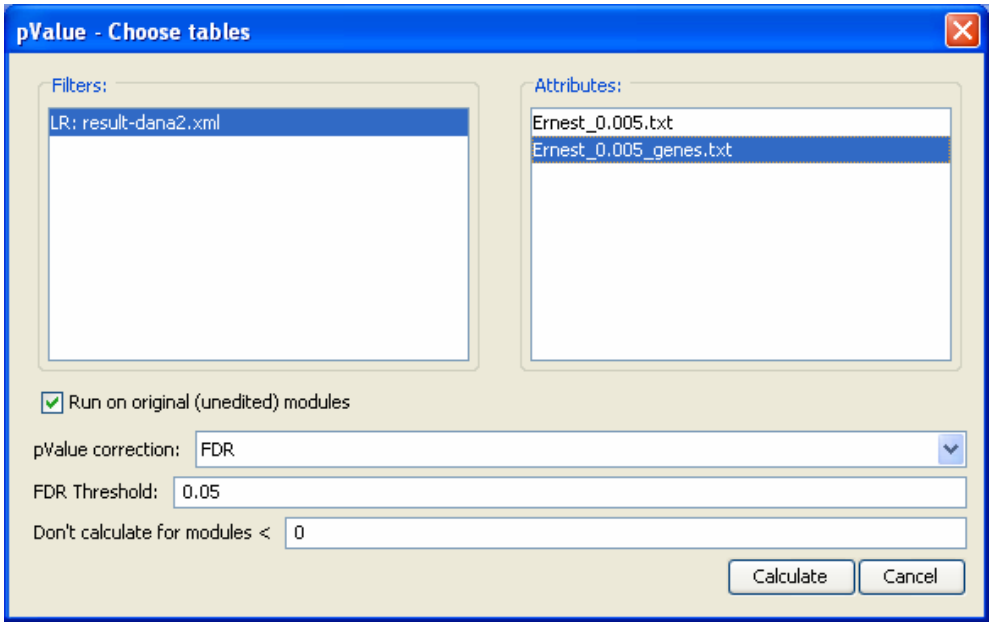


Figure 17 – p-values calculation window. Reached via Menu Project->Enrichment->Gene/Sample Attribute

In order to calculate enrichment for all modules, choose *Project->Enrichment->Gene/Sample Attributes*. The window, as shown in Figure 17, will allow you to choose the preferences for the calculation. You can choose as many filters as you like, and many attribute tables. The program will calculate all modules in the chosen filters versus all attribute in the selected attribute tables.

You can also choose to correct the p-values using FDR correction (Storey and Tibshirani 2003), currently Genatomy uses π_0=1. The corrected p-value is named *q-value* and added to the table of Figure 18.

Important information about the hypergeometric calculation:

The formula is:

$$f(k,N,m,n) = \frac{\binom{m}{k}\binom{N-m}{n-k}}{\binom{N}{n}}$$

N is the total number of genes, m is the total annotated genes, n is the set (module) size and k is the number of annotated genes inside the module.

N, the total number of genes (or features) in Genatomy is the size of main data file. For example, even if you are calculating enrichment vs. GO categories, and there are

6000 genes with GO annotations in the GO file, and you only loaded an input file (data file) with 3000 genes, N will be 3000.

Moreover, if the attribute table contains genes with missing values, the number of missing values for each attribute is subtracted from N and n.

When calculating samples enrichment for a *module network file*, Genatomy also calculates the enrichment for each of the branches, where the number of all branch samples is N, and the number of samples under one split is n.

You can also calculate hypergeometric values for two filters. In this case, one of the filters is treated as attributes (gene sets), and the other as modules. Just choose two filters without attribute tables in the *preferences window*.

Enrichment results form

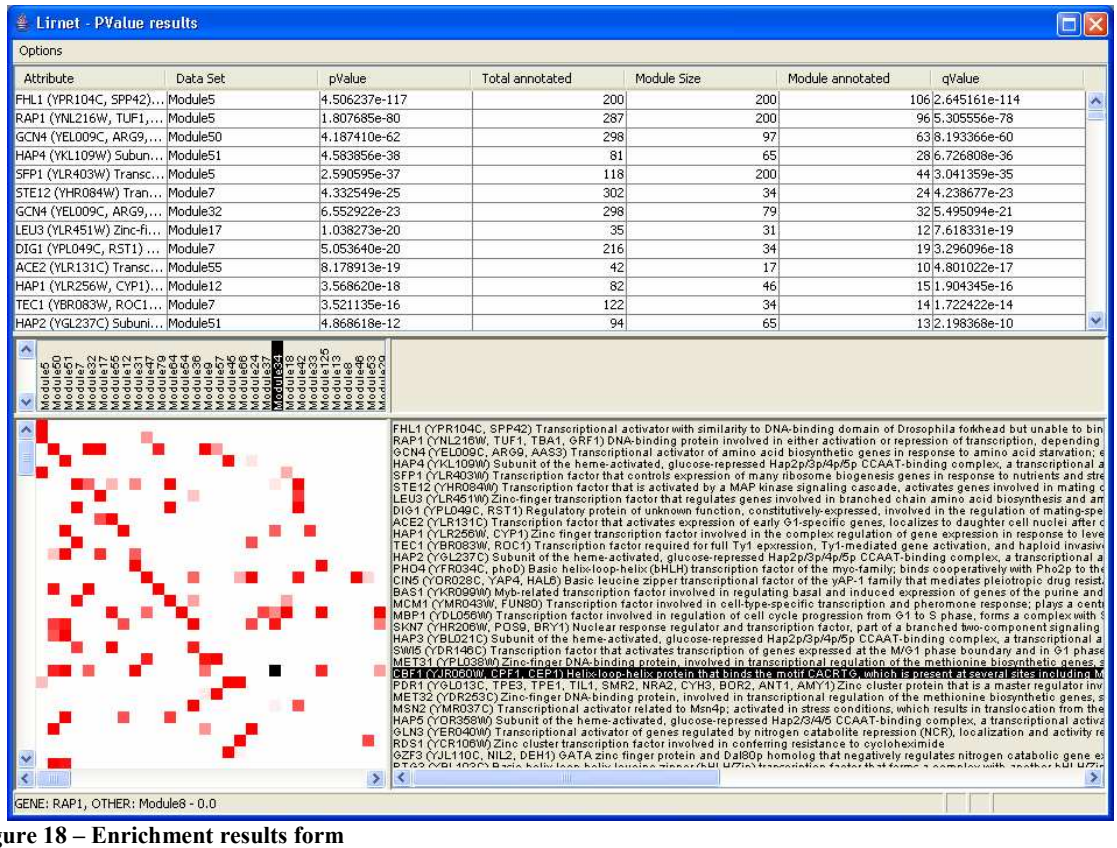


Figure 18 – Enrichment results form

The enrichment results window is divided into two sections, both showing the same results, the upper one in a table format, and the bottom one in graphical view. By double clicking on any of the results, the selected module is set in the main view, and the annotated genes and attribute are marked with a yellow background.

Genatomy also helps you compare two run results (two filter files). In the *enrichment preferences* window (Figure 17) you can select more than one filter. Each filter gets a different color in the graphical view of the results form, and you can also see a scatter plot comparing two filters (Figure 19). Each dot in the scatter plot is one attribute, and the axes are p-values for the two selected filters. If one filter is significantly better than the other, most of the dots should be on one side of the diagonal line.

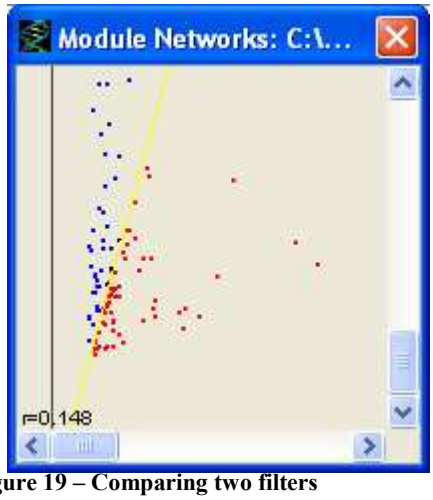


Figure 19 – Comparing two filters

Step 12: Calculating Enrichment p-values

Open the enrichment preferences window for genes (Project>-Enrichment->Genes).
Select any of the filters (modnet or LR) and GO annotation file and click RUN.
Wait a few seconds for the results window to open.
Double click on one of the results.

On-the-fly enrichment scores

As shown in Figure 20, Genatomy calculates hypergeomtric values after every change to the modules, and you can also see what the current p-values for each attribute are without opening the results form. By moving the mouse cursor over the red squares, the p-value appears on the bottom of the window.

The shades of the squares represent the p-value, where a best p-value (lower) of the module gets a fully red square, and the worst (highest) gets a white square.

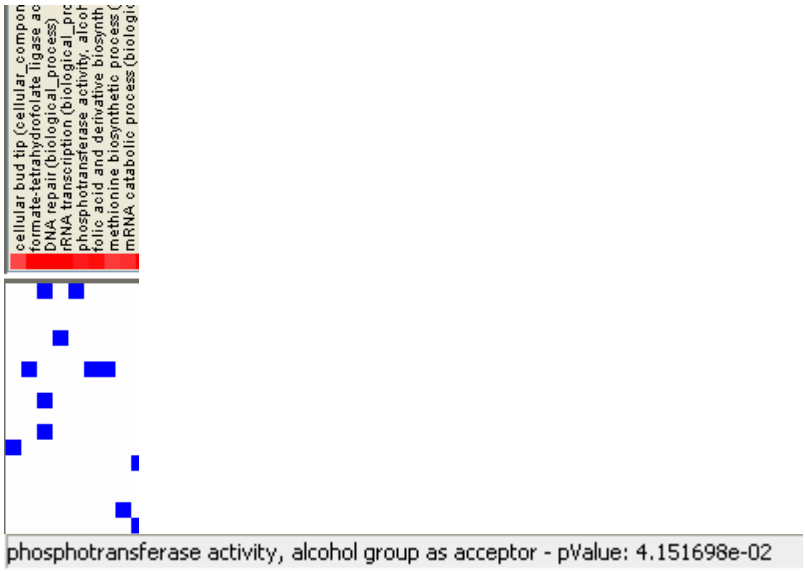


Figure 20 – On-the-fly p-values

Module Editor

Genatomy helps you make many changes to modules, and understand which genes should be in the module, and which genes should not.

When selecting Tools->Module editor, a *module editor panel*, as shown in Figure 21, appears in the window. The *module editor panel* is dynamic, and updated when a different module is selected. It contains four lists:

- In Module – All genes (or row features) currently inside the module
- Available – All genes from the main data file which aren't and weren't originally part of the module.
- Removed – Genes that originally part of the module, but were removed.
- Gene sets list – all genes set available in the project. Including Attributes sets, other modules and other sets (described later in GLSA).

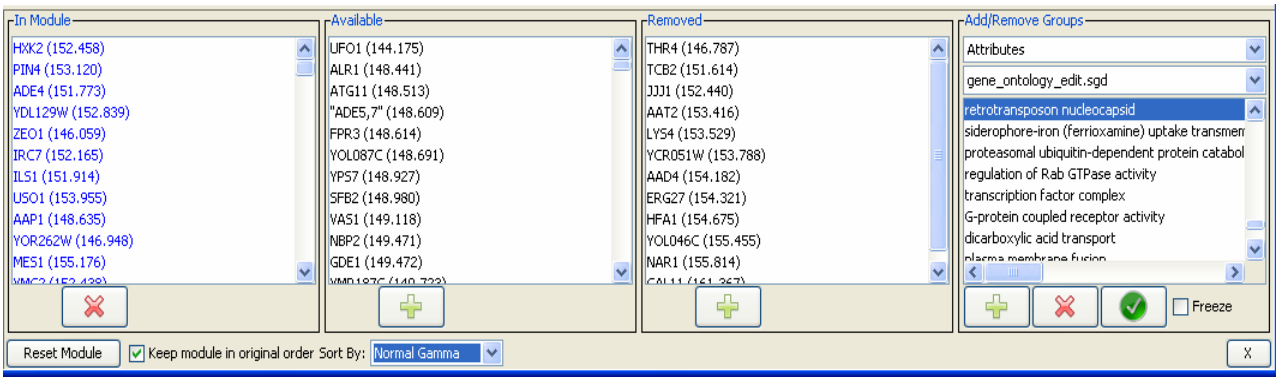


Figure 21 – Module editor panel, divided into 4 parts – currently in module genes, available genes not from the original module, removed genes and a list of gene sets.

Basic features of the panel allow you to add or remove specific genes from the module. By selecting the genes and clicking on the button in the relevant list, the genes are added or removed from the module.

Please notice – After every change, Genatomy updates hypergeometric values of the attributes, and displays the current p-values as shaded red squares next to the attribute names.

Another important feature allows you to add or remove from the module all genes from other gene sets. The forth box is the gene sets box and is different from the other three boxes. It contains three lists:

- Type of gene set – either Attribute or Filters.
- If the "Attributes" is selected in the first combo box, the second one contains all attribute tables. If "Filters" is selected, it contains all the loaded filters.
- The third list lists all the gene sets of the selected value from the second combo box. If the second combo box displays an attribute table name, the third list contains all attributes in that table. If it displays a filter, the third list shows all modules from that filter.

The buttons under those lists allow you to add or remove all genes of the selected gene set from the module. Genes will not be added twice.

The third button (the "V" icon), replaces the genes from the module with the genes from the selected gene set.

The gene set section can be controlled and changed by the user, but it is also changed whenever you double click an enrichment score in the *enrichment scores window*. The selected attribute or module will be automatically selected in the third list. If you want to freeze the list, choose "Freeze".

By default, the order of the genes is the order written in the module file, and the added genes are added at the end. The *"In Module"* list displays the original genes in blue, and added genes in black.

You can sort the genes by unchecking the box "Keep module in original order".

In the case of module network or linear regression modules, the panel also helps you to sort the genes by their agreement with the other genes in the module and the regulatory program:

- Normal Gamma – *For module networks* – Genatomy calculates the contribution of each gene to the Normal Gamma score. Smaller values mean better matching.
- Residual errors – for *linear regression modules* – the residual errors for all genes calculated using the current prediction.

The scores of the functions are displayed in parenthesis next to the gene names, and the "Available" and "Removed" lists are always sorted by that score. You can sort the current list ("In Module") as well, so the first gene is always the one with the best score.

You can always reset a module to its original gene set by clicking on "Reset".

Module Overview

Genatomy helps you review and understand algorithm results. The *module overview window* brings you a summary of the module by collecting data from all Genatomy's resources, and creates a virtual path between the modules loaded to Genatomy.

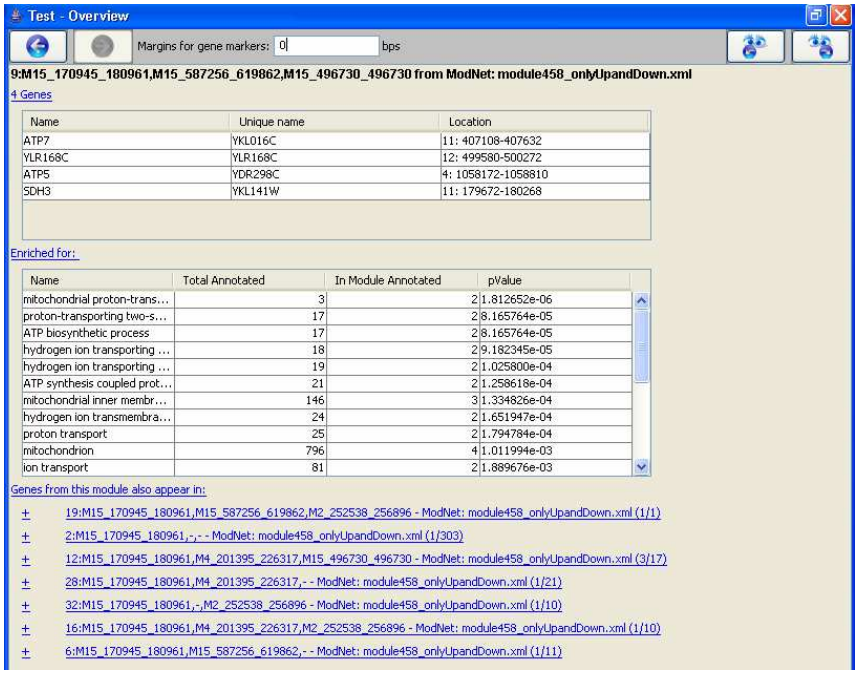


Figure 22 – Module Overview window

The overview window summarizes the results into 6 categories:

- List of the module's genes, including their genomic location.
- List of enrichment results, as calculated by Genatomy at user's request.
- List of modules that share some or all of the genes, including the overlapping genes and enrichment results.
- List of regulators. If the regulatory program contains gene markers as regulators, the window includes all genes in or around those markers as regulators.
- List of modules which contain this module's regulators as genes.
- List of modules which contain this module's regulators as regulators.

The window is updated after every change to the filters, and after enrichment run. It contains only genes from the original modules without any user changes.

The "Previous" and "Next" buttons help you navigate between modules (just like history in web browsers). The other two buttons connect the *module overview window* with the *main display*; one brings the module from the main view and reviews it while the other sets the currently reviewed module in the main view. The textbox located next to buttons controls the margins from which Genatomy extracts genes from gene markers.

Gene Lookup

Another option for reviewing results is the *gene lookup window* (Figure 23). The window lists all genes, samples and regulators in the project, and for each one it lists all modules that the feature is involved in – both as gene or sample in a module, and as a regulator.

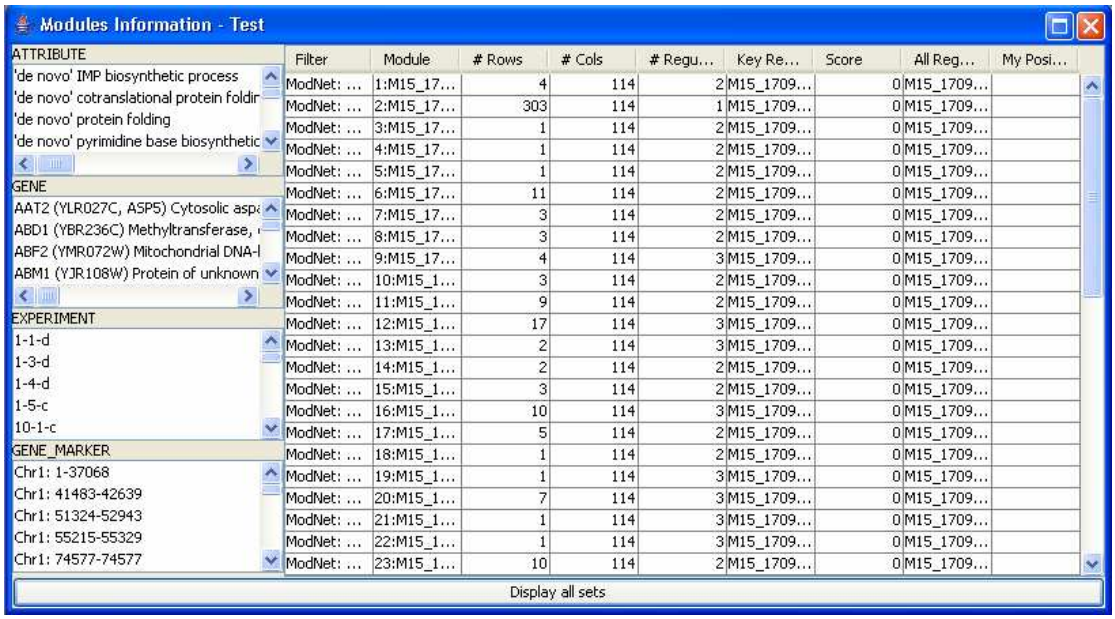


Figure 23 – Gene Lookup window

Once a module is selected, all features that are part of the module or its regulatory program are colored.

The window also summarizes important properties of modules, such as key regulators, all regulators, number of genes, etc.

Gene Interactions

Gene-interactions is one of the most useful features in Genatomy. Dana Pe'er's lab manages a gene interaction DB for each organism, incorporating public data-bases. Genatomy can query this DB and create a network of interactions, in which each node is a gene. To display the network we use Cytoscape (http://www.cytoscape.org/).

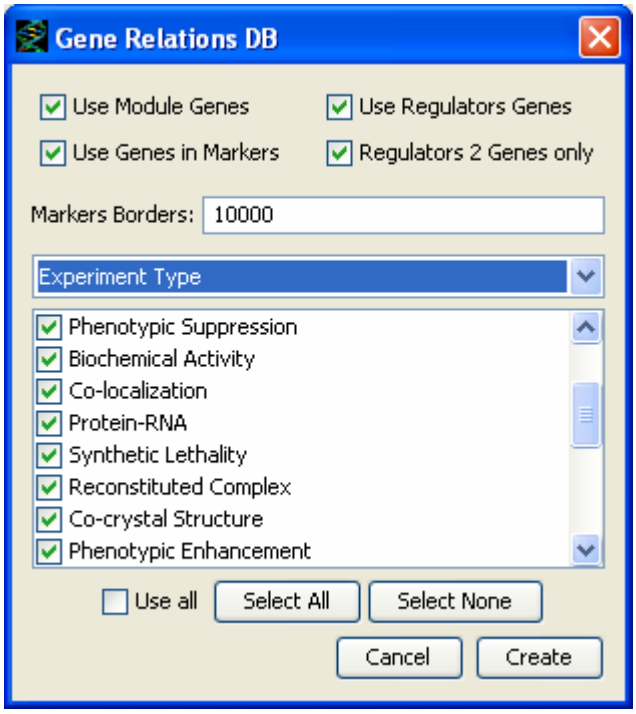


Figure 24 – Gene Interaction properties window

In order to create the network, go to Project->Gene interactions, choose from the available options and click on Create (Figure 24). Many files are created during the query, and all are essential for display the network in Cytoscape. Genatomy asks you to choose an output directory, and opens Cytoscape automatically. If it fails to opens Cytoscape, it leaves a file named – *commandline*, with a command to open and load the last created network in Cytoscape.

This feature only works on the current selected module. The properties window, as shown in Figure 24, allows you to select many features of the network:

First, you can select which genes you want to see in the network:

- Module Genes
- Regulatory Genes
- Gene inside markers (if any exist as regulators)

You can also select to query interactions between module genes and regulatory genes only (Regulators 2 Genes only).

In case you choose to include genes from genes markers, you can choose the margins in which the genes are extracted.

The combo box and the following list allow you to choose the sources and the types of interactions.

Genatomy extracts important information about the genes to Cytoscape, including their origin (part of module/regulators), their gene marker (if extracted from one), unique ID and name, and other information about the genes.

To see them all select to see all node attributes in Cytoscape.

The DB is also accessible via WSDL client. For more information contact Dana Pe'er's group.

Bird's Eye view

Another simple and yet useful feature is *Bird's Eye View*, accessible via the menu Tools->Bird's Eye View.

This window displays all modules of a selected filter sequentially, allowing you to see an overview of the modules.

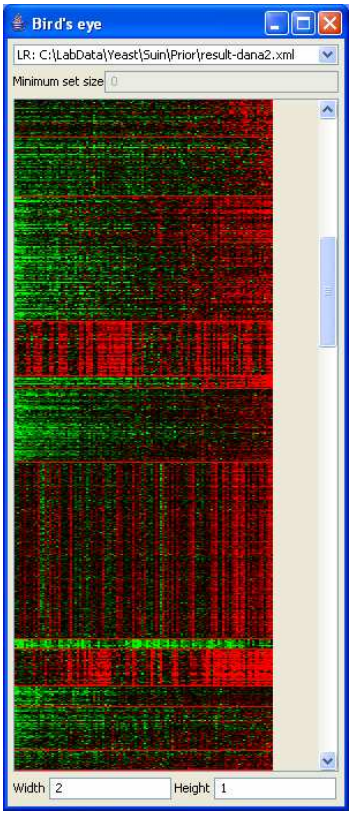


Figure 25 – Birds Eye View

GLSA – Genome Location Set Analysis

GLSA is an on-working algorithm, an expansion of GSEA (Subramanian, Tamayo et al. 2005). It searches inside modules for several genes in a small genomic region.

The algorithm scores the genomic distribution of a module's genes, one score for each chromosome, and performs a permutation test to identify statistically significant regions.

In order to run the algorithm, choose Project->GLSA, select the properties and RUN. Please notice, the algorithm takes time to run since it performs permutation tests, but the results are saved and loaded with the project.

The algorithm allows you to choose the weights of genes included in the set (module) and not in the module, and the constants N_k and N_r. It does not allow the region to include a centrosome.

The *results window* looks very much a like the *hypergeometric results window*, and displays the significant genomic region, its length and the genes in it. A double click will set the module in the main view, and will set the results as an ad-hoc gene set in the *module editor view*, so you can also set the module to include only the genes from the region in their order.

For more information about the algorithm please contact Dana Pe'er's group.

Other options

Coloring Methods

There are 3 coloring methods in Genatomy:

- Gradient
- Discrete
- Ranges

Each data panel (e.g. *Module networks panel*) has its own coloring scheme, so genes in the main view and genes in other panels are not colored with the same scheme. For example, genes in the main panel can be painted with a *gradient method* and colored with red and green, while gene in *Module networks panel* can be colored with blue and yellow.

In some cases, you can choose which coloring method to use. *The coloring configuration panel* on the *side bar* (Figure 3) has a combo box with all three coloring methods. By choosing and clicking on replace, the coloring method for that data type will be replaced.

The properties of *Gradient coloring method*s are described above.

In *Discrete coloring method*, each value in the data has its own color, up to 20 values. If the data contains more than 20 different values, the color scheme will color everything in black, and will request you to switch a coloring method.

The *ranges coloring methods* is constructed by ranges of values, in the form of bigger or smaller than constants, each range with a different color. The user can determine the ranges using a *configuration panel*, as shown in Figure 26.

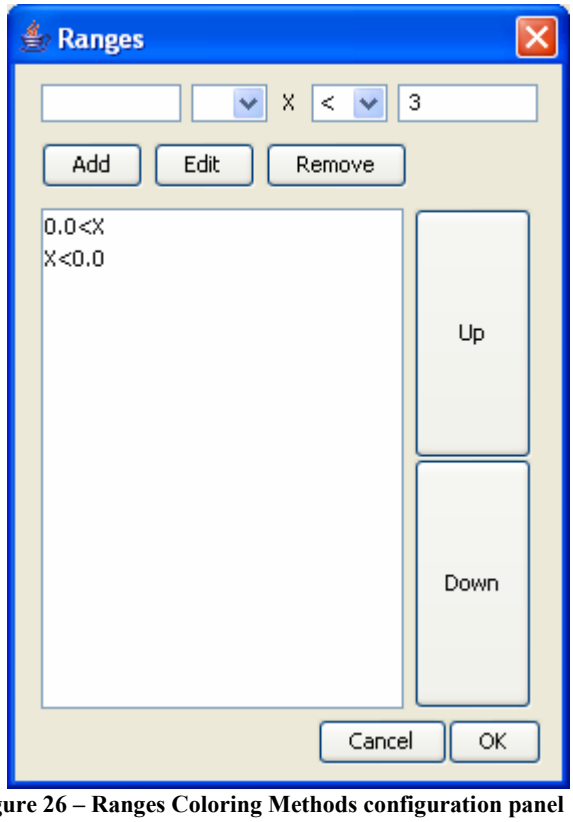


Figure 26 – Ranges Coloring Methods configuration panel

Find

Genatomy has a common feature in all programs – Find. Just click on CTRL+F or edit->find and the search tool bar will appear under the *main display*. The bar looks and acts just like the search function in Firefox. It searches in all panels displaying text, in a circular order, starting with *sample names panel*.

Export

Genatomy can export the displayed data, both tables and figures, by choosing File->Export->Method.

The figures export function saves the figures in the resolution currently on the screen, so a 10 pixel square in on screen will be saved as 10 pixels in the file. If the picture is too big, decrease the dimension of the screen display.

Importing GXP

If you have never worked with Eran Segal's visualizer called Genomica, you might want to skip this part. If you do have GXP files generated by Genomica, Genatomy can import them.

Choose File->Import->GXP and start the importing process. You will be asked to choose an output directory. Unlike Genomica that packs all the data into one XML file, Genatomy splits the data into several files: Main file (expression), attribute files (annotations), genes information and filter files (*Module network* results). After importing, all the above files will be created in the output directory in addition to a project file (GPF), which you can use to open this project again.

After selecting an output directory, you will choose the species.

If the file contains more than one gene name column (as are most GXPs), then you will be asked to choose which column should be used as the name. You will also have the option to use the other columns as aliases and description for the genes. Since Genatomy can load updated data for all genes, it is highly recommended to avoid doing so. In most cases, Genatomy can add this information using the repository.

Importing GXA

Genatomy can import GXA (Genomica's attributes file). When a project is open, you can select File->Import->Import GXA then choose a destination folder. The GXA will be printed as a tab-delimited file, and will be added automatically to the attributes manager.

You only need to import a file once, and you can use the new file that Genatomy creates from now on.

File formats

Expression and Attribute files

All the files Genatomy reads must be tab-delimited files, meaning that all lists of names and/or values are separated by a "TAB" ("\t").

Expression files or any other data files in the format of columns, rows and values, starts with a TAB, then a list of columns names (separated by TAB), end-of-line, and each other line starts with a name and then a list of values.

If there are missing values, they will be interpreted as MISSING DATA.

Attached is an example for an expression file:

	1-1-d	1-3-d	1-4-d	1-5-c	2-2-d	2-3-d
SCC2	-0.770102257	-1.134186598	0.598854867
FLO1	-0.748873771	0.142137332	-0.975609746
MYO3	0.153207965	-0.896643339	-1.083990541
PDR10	0.11107875	-0.646593365	-0.796089208
SCP160	-0.980517144	-0.572738143	1.208820598
TSL1	1.098976424	-0.506527311	-1.21852392
YJR041C	-1.062371373	0.245006437	0.707946776
HAP1	0.022874054	-0.426239207	0.42324634
NET1	-0.833998425	-0.225989518	0.449110028
GCN1	-0.815871471	-0.779669538	0.503853536
RAV1	-0.65721084	-0.891468418	-0.851079181
ERG1	-0.711441245	-1.129265354	-0.660373854

The first row contains the column names, starting with a tab. The first column, starting in the second row, is the gene names. The rest of the file is filled with values.

Naming

As mentioned before, when Genatomy reads a name it selects the type of the data (gene, attribute, gene marker, see Table 1 for more), by the context in which the name was read.

By default, columns of expression file are samples, columns of attribute files are attributes, rows of sample attribute file are samples, and rows of expression files are genes (or the default type chosen by the user in *New Project Form*, Figure 2).

If you want Genatomy to treat a name in a way other than the default, you need to specify it.

By adding the desired type next to the name after a coma, e.g. ORF0001W, GENE, you tell Genatomy how to treat the feature. For example, if you are loading an attribute table, describing knockout results for genes, you can name the columns by the knocked out gene, and the display will add information about the genes in all places that the attribute name appears.

The available types are listed in Table 1.

You don't need to add the type in filter files.

Filters

Simple Sets file

An XML file. The skeleton of the file is:

```
<Root>
        <Module ….>
        </Module…>
        <Module ….>
        </Module…>
        …….
</Root>
```

Each module has this structure:

```
<Module Name="My Name">
<Set>
GENE1\tGENE2        GENE3
</Set>
<Samples>
Sample1\tSample2      Sample4
</Samples>
</Module>
```

A module that does not contain Set or Samples nodes will be loaded with all genes or samples listed in the main data file.

The order of the genes/samples determines the order in which the module will be displayed.

Module Network

Module network file has the basic structure of a Simple Set file, with additional information describing the regulatory program.

The basic structure of a module is:

```
<Module Name="My Name">
<Set>GENE1\tGENE2          GENE3</Set>
<Samples>Sample1\tSample2          Sample4</Samples>
<Split Type="OneRegulator">
<SplitData>described below</SplitData>
<Left>
        <Split>same structure of main split</Split>
</Left>
<Right>
        <Split>same structure of main split>
</Right>
</Split>
</Module>
```

Each split node has the same basic structure, where child nodes (left and right splits) are listed in the left and right nodes (a recursive format). If a split is a leaf, or has only one child (left or right), the node can specify only (or none).

A split node has the next structure:

```
<Split Type="OneRegulator">
<SplitData Regaultor="RegulatorName"
SplitPoint="SplitValue" Type="less"/>
<Left>
        <Split>same structure of main split</Split>
</Left>
<Right>
        <Split>same structure of main split>
</Right>
</Split>
```

Where RegulatorName is the name of the regulator (either gene or attribute name), and SplitValue is the value in which the branch is split (must be a number).

Linear Regression

Linear regression file has the basic structure of a Simple Set file, with additional information describing the regulatory program.

The structure of a module is:

```
<Module Name="My Name">
<Set>GENE1\tGENE2          GENE3</Set>
<Samples>Sample1\tSample2          Sample4</Samples>
<Regulators>Reg1\tReg2       Reg3</Regulators>
<coefficients>0\t1       0.2       0.4</coefficients>
</Module>
```

The "coefficients" node lists the coefficients of the listed regulators. If the node lists more than the listed regulators, the last coefficient is interpreted as intercept.

The node doesn't have to appear, and if it does not, Genatomy calculates a least squares fitting automatically.

Information files

Conversion table

Conversion table is a file, containing two columns separated by a tab. Each line contains two names. Genatomy will create a list of names for each gene, so if a name repeats more than once in the file, Genatomy will use all pairs to create the list.

Description Table

A file contains descriptions for genes, where each line starts with a name, separated by tab follows the description.

Full genome list

A full genome list contains all the information about the genes, including unique identifier, name, location and description.

For some organisms, a similar file can be found in one of the organism's web site. For others, the file has to be constructed from several sources.

The "*full genome file*" format is different from one organism to the other, and for organisms with an existing file, Genatomy's format is the same as the published file.

S. cerevisiea

Sgd_features file from http://www.yeastgenome.org/

Human

Genatomy's format, tab-delimited file. The columns are:

```
UID\tName\tOtherNames\tChromosome\tStrand\tFirst BP\tLast BP\tDescription
```

Each line corresponds to one gene. If information is missing, an empty spot is left. If there are several "Other Names", they should be separated by a "|" (horizontal bar).

Strand values should be "W" or "C".

Chromosome can be number (for 1-23), X or Y.

References

Segal, E., M. Shapira, et al. (2003). "Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data." Nat Genet **34**(2): 166-76.

Storey, J. D. and R. Tibshirani (2003). "Statistical significance for genomewide studies." Proc Natl Acad Sci U S A **100**(16): 9440-5.

Subramanian, A., P. Tamayo, et al. (2005). "Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles." Proc Natl Acad Sci U S A **102**(43): 15545-50.