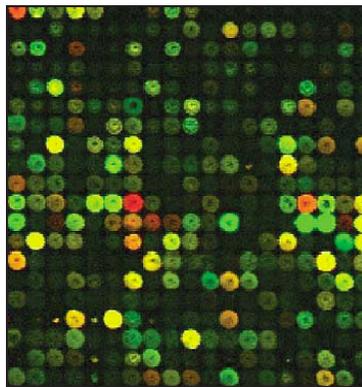


MicroArray Genome Imaging and Clustering Tool
MAGIC Tool Instructor's Guide



June 4, 2003

Updated November 13, 2006

Educational Goals

Bring Hands on genomics to undergraduates at no cost.

Teach students how DNA microarray data are generated and analyzed.

Let students perform the steps required to analyze microarray experimental data.

Let students discover sources for variation within DNA microarray analysis.

Segue from simulations to real data analysis, clustering and exploring.

Provide students with Discovery Questions that guide students through data analysis in a question-driven format.

Allow students to compare discovery science vs. hypothesis testing.

Student Audience

The tools provided in this workshop would be appropriate for students taking sophomore level genetics or higher courses. Students need to understand the central dogma of gene activation.

Students should be familiar with cDNA and its production. Familiarity with methods such as DNA sequencing and Southern blots would be beneficial.

Students taking genetics, molecular biology, cell biology, development, biochemistry, genomics, computational biology, physiology, neurobiology, etc. might be well suited for this module.

Time Frame

Working with DNA microarray data could be chopped up into different sized lessons. You could combine online resources with MAGIC Tool to give students a overview of the method. On the other extreme, you could use MAGIC Tool with research students to mine public databases and/or your own microarray data. In short, this module can be tailored to suit your needs.

Flexibility for Starting Places

MAGIC allows you to import tiff files, ratios, or Affymetrix data for clustering and exploration.

You can use GCAT tiff files as a source for raw data and student exercises.

www.bio.davidson.edu/GCAT

System Requirements:

Windows 2000 or later

Mac OSX 10.2 or later

Linux 7.x or later

256 MB RAM minimum; 384 MB to 1 GB of RAM recommended.

Disclaimer: Laurie Heyer and I wrote a textbook called *Discovering Genomics, Proteomics and Bioinformatics* which was published jointly by Benjamin Cummings and Cold Spring Harbor Laboratory Press. One topic covered in this book is DNA microarrays and therefore, I am partial to the way we covered DNA microarrays in our book. However, you may use all of the course materials from the web site free of charge <www.awl.com/genomics>.

Introduction

Overview

Now that whole genomes are sequenced, how can we use this information to understand biology better? What would be really nice is if we could measure simultaneously the activity of every gene in the genome. However, nobody wants to produce 6,200 Northern blots to measure every gene's activity in yeast, not to mention 14,000 for *Drosophila* or 35,000 for humans! What we need, is a device that will allow us to perform the equivalent of a Northern blot but do this simultaneously for every gene in the genome.

Behold! DNA microarrays are here. DNA microarrays were invented in the 1990's and in less than ten years the method has grown from a cottage industry to an international billion dollar market (see 11 April, 2003 *Science*, pp 236-239). DNA microarrays allow investigators to compare whole-genome gene expression from two samples. The best way to understand how this method works is to view an animation. Go to this URL to learn how this method is performed <www.bio.davidson.edu/courses/genomics/chip/chip.html>. To see how DNA microarrays are printed, you can visit this page <www.bio.davidson.edu/courses/genomics/arrays/arrayer.html>. In short, each gene in the genome is spotted onto a glass microscope slide and immobilized. mRNA is isolated from cells and tagged with fluorescent dye. The fluorescent probe is incubated with the spotted slide and allowed to base pair. The dye is detected on the slide and this signal indicates which genes were active and how much mRNA was produced. An overview can be seen at this URL <www.bio.davidson.edu/people/maccampbell/strategies/chipsintro.html>.

Experimental Design

The most popular form of DNA microarrays requires the investigator to produce two sets of fluorescently-labeled cDNA probes (typically green and red fluorescent dyes are used). [The term probe is somewhat controversial in this field. Some people think the labeled cDNA should be called target and the spotted DNA should be called probe, since only the spotted DNA is a known sequence. However, we will call the labeled cDNA probe and the bound DNA on the glass will be called target.] Control set of cDNA probes was generated from cells grown under standard conditions. The experimental set of cDNA probes was produced from cells grown under the condition under investigation. With these two sets of probes, the investigator can determine RELATIVE expression levels for every gene in the genome.

Using this basic design, investigators can compare differential gene expression due to different growth conditions, cancerous tissue, drug response, viral infection, development, and an unlimited number of additional conditions. The only limitation is the creativity of the investigators.

Data Processing

Raw Data

When a DNA microarray experiment is performed, the raw data come in the form of two picture files. The format of the file is called a TIFF file and the suffix is ".tif". These are gray

scale images with the data from each color dye captured in a separate file (green.tif and red.tif). Specially designed software takes these two tiff files and generates a color associated with the different intensities of gray. If red and green dyes are colocalized, the software will depict a yellow color. In this way, we are able to see the typical green, yellow, red DNA microarray images of raw data (figure 1).

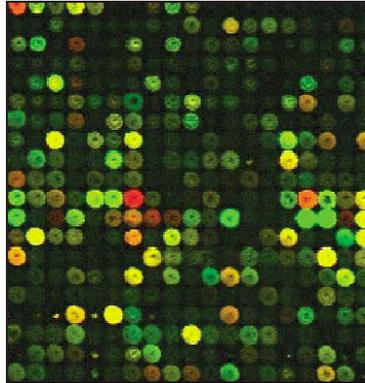


Figure 1. DNA microarray raw data. (from: 11 April, 2003 *Science*, pp 236-239)

Quantifying Spots

The next step in the process is to convert the tiff files into numerical values. First, the software must locate the spots (figure 2), and then distinguish signal from background (figure 3). MAGIC Tool can perform both of these steps.

Figure 2. Locating each spot using MAGIC Tool's unique three click method.

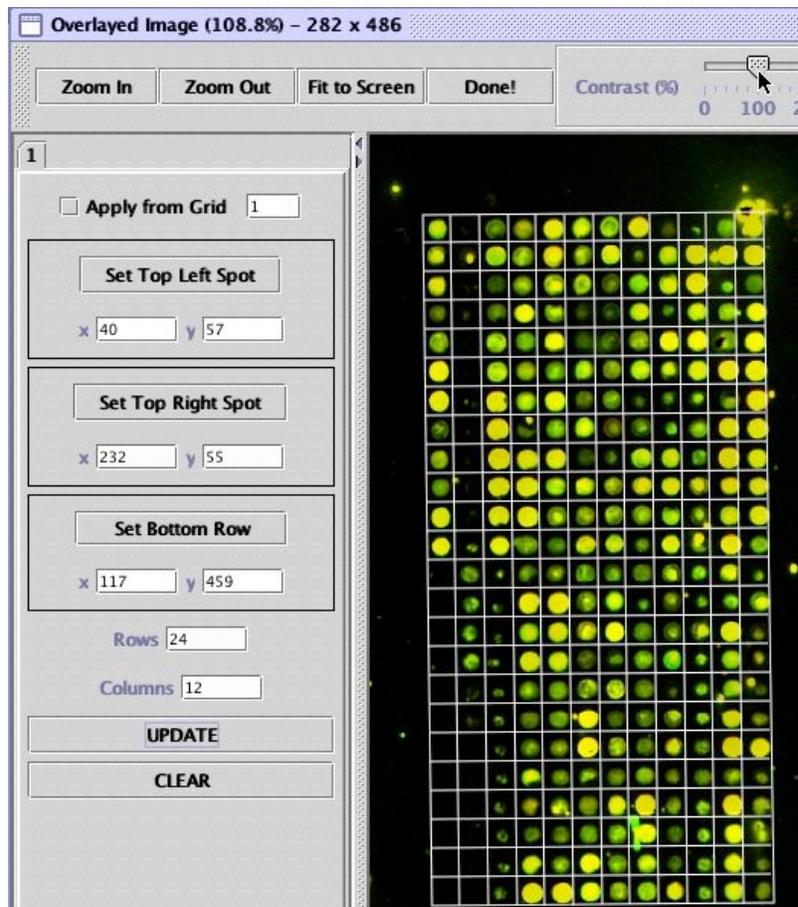
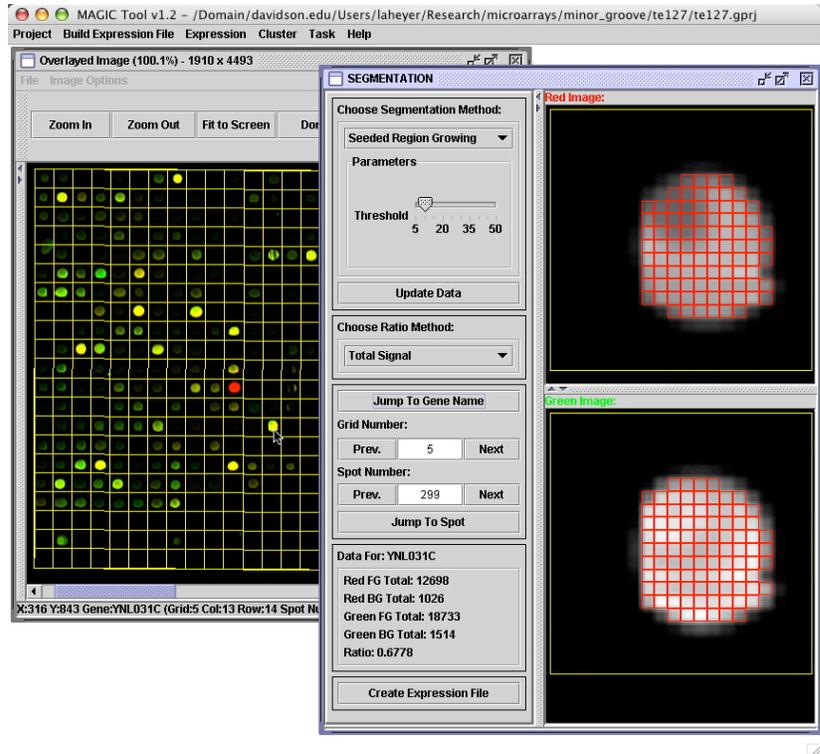


Figure 3

Distinguishing signal from background. MAGIC Tool can use three different methods to determine which pixels are signal and which pixels are background. Shown here is MAGIC Tool's implementation of the seeded-region growing method.

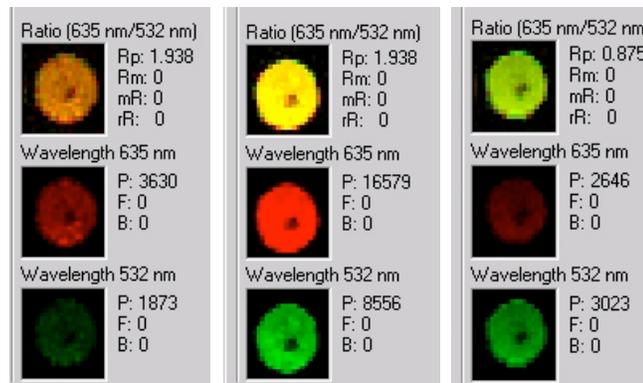


Once the software knows where to locate the florescent dye data, it quantifies the amount of red and green present in the two tiff files (figure 4).

These numbers are then used to calculate ratios of expression. Often, the control cDNA is labeled green and the experimental cDNA is labeled red. Using this convention, then the numbers for red (experimental) are divided by their corresponding green numbers (control). This quotient indicates which genes are induced (more red dye bound to a spot than green dye and a ratio greater than one) and which genes are repressed (more green dye bound to a spot than red dye and a ratio less than one).

Figure 4.

Comparing expression levels for three different genes.



Let's look at three different spots from one experiment. The gene on the left was induced and thus appears more red (middle) than green (bottom) and appears orange (top); it would have a ratio greater than one. The middle gene was transcribed equally well under both conditions (middle and bottom) and thus appears yellow (top); it would have a ratio very close to one. The right gene was repressed since more green cDNA bound (bottom) than red cDNA (middle) so its overall color is lime colored (top); its ratio would be less than one.

One last operation needs to be conducted by the software. We need to match the spot coordinates with the gene names. Since we printed known DNA at known locations, we need to have a spreadsheet file that can be interpreted by the software to identify each spot. As you can imagine, this is a critical step for subsequent data analysis. It is very easy to get confused or lost on your massive DNA microarray, so a lot of care has to be taken at this time. Once you know your way around the microarray, you can deduce the gene list pattern and make sure you name each spot appropriately.

Comparing and Clustering Genes

Up to now, the conversion of mRNA levels to numbers has been fairly intuitive. Now we need to move into a more mathematical mode so we can begin to compare genes to each other. We need to go through three steps in order to extract biologically meaningful information from these ratios. First, we must level the playing field so that a 4 fold induction and repression are on the same scale. Second, we need to compare expression levels between all gene pairs. Third, we need to cluster genes together that behave similarly. We will address each of these steps in turn.

Converting ratios to avoid fractions

Intuitively, we know that if a gene changes from 4-fold induced to 16-fold induced, this should be an equivalent change to a gene changing from 4-fold to 16-fold repressed. However, if we take the corresponding ratios at face value (4 and 16 for induction vs. 0.25 and 0.0625 for repression) the changes do not appear equivalent. This can be seen easily by graphing them. For example, consider the ratios for 12 hypothetical genes, measured at 6 different time points, in Table 1. These gene expression profiles are graphed in Figure 5.

Table 1: Ratio data showing fold change (experimental ÷ control) in mRNA production for 12 hypothetical genes (C – N).

Name	0 hours	2 hours	4 hours	6 hours	8 hours	10 hours
gene C	1	8	12	16	12	8
gene D	1	3	4	4	3	2
gene E	1	4	8	8	8	8
gene F	1	1	1	0.25	0.25	0.1
gene G	1	2	3	4	3	2
gene H	1	0.5	0.33	0.25	0.33	0.5
gene I	1	4	8	4	1	0.5
gene J	1	2	1	2	1	2
gene K	1	1	1	1	3	3
gene L	1	2	3	4	3	2
gene M	1	0.33	0.25	0.25	0.33	0.5
gene N	1	0.125	0.0833	0.0625	0.0833	0.125

We need a way to level the playing field and the best way to do this is to log transform the ratios. For example, if you take the log base 2 of 0.0625 and 16:

$$\log_2(16) = 4$$

$$\log_2(0.0625) = \log_2(1/16) = \log_2(1) - \log_2(16) = -\log_2(16) = -4$$

As shown in Figure 5, using a log transformation makes a 16 fold induction and a 16 fold repression appear to be equal in scale but opposite in direction (4 and -4). A summary of the log transformation of all 12 gene profiles is shown in Table 2.

Figure 5

Ratios from Table 1, before and after log₂ transformation. Genes that are 16-fold induced and 16-fold repressed at 6 hours are highlighted in red.

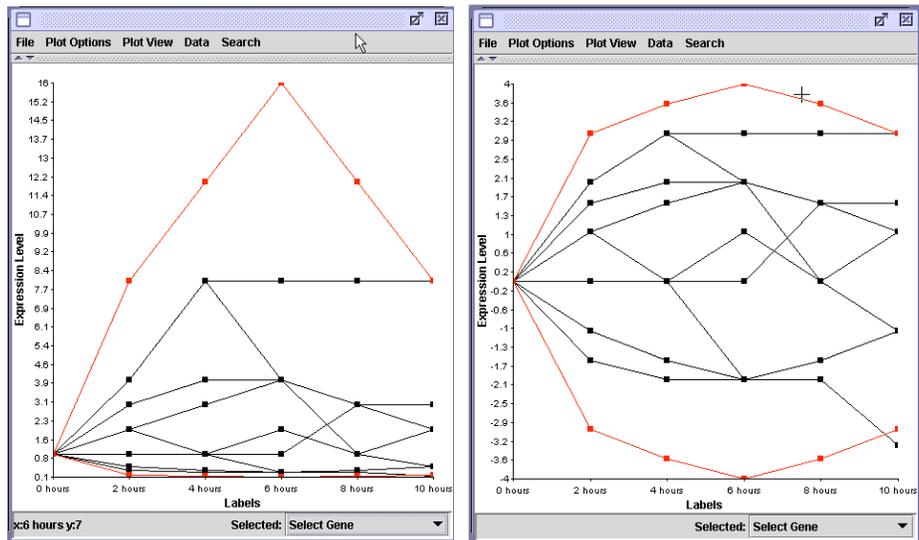


Table 2: log₂ transformation of gene expression data in Table 1

Name	0 hours	2 hours	4 hours	6 hours	8 hours	10 hours
gene C	0	3	3.58	4	3.58	3
gene D	0	1.58	2	2	1.58	1
gene E	0	2	3	3	3	3
gene F	0	0	0	-2	-2	-3.32
gene G	0	1	1.58	2	1.58	1
gene H	0	-1	-1.60	-2	-1.60	-1
gene I	0	2	3	2	0	-1
gene J	0	1	0	1	0	1
gene K	0	0	0	0	1.58	1.58
gene L	0	1	1.58	2	1.58	1
gene M	0	-1.60	-2	-2	-1.60	-1
gene N	0	-3	-3.59	-4	-3.59	-3

Comparing genes

What we would like to do is find genes that alter their expression patterns in similar ways, even if the magnitude of the mRNA production is not equivalent. To do this, we will need to calculate the Pearson **correlation coefficients** for each gene pair. A correlation of 1 means two genes had perfectly synchronized expression profiles (both were repressed or induced in a coordinated manner). A correlation of -1 means two genes had exactly opposite expression profiles (one was induced while the other was repressed). A correlation near 0 means that two genes had no similarities in either induction or repression.

To find the correlation between genes D and L, $r(D,L)$, first compute the mean and sample standard deviation of the expression values for each gene (i.e. each row in Table 2):

$$\bar{X}_D \approx 1.36 \quad \bar{X}_L = 1.19 \quad s_D \approx 0.762 \quad s_L \approx 0.701.$$

Subtract \bar{X}_D from each value in the D row and divide each result by s_D . The result is a row of **normalized** values in the D row:

$$D_{\text{norm}} = -1.788, 0.293, 0.838, 0.838, 0.293, -0.475.$$

Do the same in the L row, this time subtracting \bar{X}_L and dividing by s_L to produce the following normalized row:

$$L_{\text{norm}} = -1.705, -0.278, 0.556, 1.149, 0.556, -0.278.$$

Now multiply the first number in D_{norm} by the first number in L_{norm} , the second number in D_{norm} by the second number in L_{norm} , and so on, keeping a running sum of these products. Finally, divide this sum (4.691) by **one fewer than** the number of elements in each row (6 time points – 1 = 5) to get the correlation coefficient $r(D,L) = 0.938$. Note that Excel has a built-in function CORREL that you can use to compute correlation coefficients. Table 3 summarizes the correlation coefficients for every possible pair of genes. **It is very important to log-transform ratios before computing correlation coefficients.**

Table 3: Correlation coefficient between each pair of genes, based on \log_2 -transformed gene expression data in Table 2.

	gene C	gene D	gene E	gene F	gene G	gene H	gene I	gene J	gene K	gene L	gene M	gene N
gene C	1	0.94	0.96	-0.40	0.95	-0.95	0.41	0.36	0.23	0.95	-0.94	-1
gene D	0.94	1	0.84	-0.10	0.94	-0.94	0.68	0.24	-0.07	0.94	-1	-0.94
gene E	0.96	0.84	1	-0.57	0.89	-0.89	0.21	0.30	0.43	0.89	-0.84	-0.96
gene F	-0.40	-0.10	-0.57	1	-0.35	0.35	0.60	-0.43	-0.79	-0.35	0.10	0.40
gene G	0.95	0.94	0.89	-0.35	1	-1	0.48	0.22	0.11	1	-0.94	-0.95
gene H	-0.95	-0.94	-0.89	0.35	-1	1	-0.48	-0.21	-0.11	-1	0.94	0.95
gene I	0.41	0.68	0.21	0.60	0.48	-0.48	1	0	-0.75	0.48	-0.68	-0.41
gene J	0.36	0.24	0.30	-0.43	0.22	-0.21	0	1	0	0.22	-0.24	-0.36
gene K	0.23	-0.07	0.43	-0.79	0.11	-0.11	-0.75	0	1	0.11	0.07	-0.23
gene L	0.95	0.94	0.89	-0.35	1	-1	0.48	0.22	0.11	1	-0.94	-0.95
gene M	-0.94	-1	-0.84	0.10	-0.94	0.94	-0.68	-0.24	0.07	-0.94	1	0.94
gene N	-1	-0.94	-0.96	0.40	-0.95	0.95	-0.41	-0.36	-0.23	-0.95	0.94	1

Alternative Methods

With MAGIC Tool, you can calculate dissimilarities three different ways. The first and most common way is $d = 1 - \text{correlation}$. The reason correlation is subtracted from 1 is to turn it into a distance. When the expression profiles of two genes are perfectly correlated (correlation = 1), the dissimilarity is 0 ($d = 1 - 1 = 0$). When the expression profiles of two genes are exactly opposite, the dissimilarity is 2 ($d = 1 - (-1) = 2$). Therefore, the possible values for dissimilarity are between 0 and 2. Two other methods for computing dissimilarity are l^p and Jackknife correlation.

l^p and Euclidean distance

Euclidean distance is the usual distance formula for the distance between two points. In two dimensions, the Euclidean distance between (x_1, y_1) and (x_2, y_2) is $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. In multi-dimensional space, the notation is a little messier, but the idea is the same. For example, in n dimensions, a point (e.g., gene expression over n timepoints) is specified by $g = (g_1, g_2, \dots, g_n)$ and the Euclidean distance between g and h is $\sqrt{(h_1 - g_1)^2 + (h_2 - g_2)^2 + \dots + (h_n - g_n)^2}$. The l^p distance is a generalization of the Euclidean distance in which the power on each term can be any positive integer p , and the root is the corresponding one. In other words, the Euclidean distance is the l^p distance with $p = 2$.

Jackknife correlation

Falsely high correlation values may result due to outliers. The jackknife correlation detects and avoids falsely high correlation coefficients due to a single outlier by taking the minimum correlation value. Jackknife correlation computes the correlation with one time point deleted, repeating for each and every time point, and takes the minimum of all these correlation coefficients. The mathematical notation is: for a pair of genes, i, j , let ρ_{ij} represent the correlation coefficient for the pair. Then let $\rho_{ij}^{(k)}$ represent the correlation coefficient with the k^{th} time point deleted. Then, the jackknife correlation of two gene expression vectors with t time points is given by $J_{ij} = \min\{\rho_{ij}^{(1)}, \rho_{ij}^{(2)}, \dots, \rho_{ij}^{(t)}, \rho_{ij}\}$. The jackknife correlation is much more conservative in assigning similarity than the usual correlation. However, it takes considerably longer to compute. Also, if the data contains many outliers across several time points, jackknife correlation may not detect false positives effectively.

Grouping similar genes

Now that we have compared every gene against every other gene, we would like to cluster similar genes next to each other. There are several methods for doing this and MAGIC Tool can perform 4 different clustering methods. The most common, but not necessarily the best, method is **hierarchical clustering**.

Hierarchical clustering works as follows. Find the two most similar genes in the entire set of genes. Join these together into a cluster. Now join the next two most similar objects, where an object can be a gene or a cluster. To do this, you need a method for measuring the similarity of each remaining gene to the first cluster of two genes. One way is to average the log-transformed expression patterns of the two genes in the cluster, forming a new expression pattern that

represents the cluster of two genes. All remaining genes can be compared to this new pattern by computing the correlation coefficient of each one with the pattern. Continue this process, joining objects in the order of their similarity to one another, until all genes in the original set have been clustered.

Let's walk through the process to cluster the genes in table 3. First, find the two most similar genes in the entire set of genes. Genes L and G are the most similar since $r(L,G) = 1$. Join these together into a cluster, denoted [LG]. Cluster [LG] is added to the list of available objects and the single genes L and G are removed from the list. Now join the next two most similar objects, using the procedure described above. (Note that in this case, the average of L and G is equal to both L and G, so we are saved the job of computing new correlations.) The closest gene to the cluster [LG] is gene C, since $r(C,G) = r(C,L) = 0.95$. However, gene C and cluster [LG] are not the two most similar objects; rather genes C and E are, with $r(C,E) = 0.96$. Thus we join genes E and C to form cluster [EC].

At the next iteration, we need to know the correlation of each gene with the average log transformed expression patterns of genes E and C: 0, 2.5, 3.29, 3.5, 3.29, 3. The correlations with this pattern for genes not yet clustered are as follows:

D	F	H	I	J	K	M	N	[LG]
0.90	-0.48	-0.93	0.32	0.33	0.32	-0.90	-0.99	0.93

The most similar gene to [EC] is gene D. Gene D is even more similar to [LG], since $r(D,G) = 0.94$. But the two most similar objects now are genes N and H, with $r(N,H) = 0.95$. Therefore, we join genes N and H to form cluster [NH]. As you can see, each time two objects are clustered, a new set of correlation coefficients needs to be produced to take into account any previously clustered objects. This is a number crunching process and one ideally suited for computers, not humans.

The hierarchical clustering process for these 12 genes is summarized in the following table. Note that the final object that is created is the clustering of all 12 genes shown in Table 4. You can learn more about clustering at this interactive web page:

<http://gcat.davidson.edu/DGPB/clust/clustering.htm>

Table 4: Summary of the hierarchical clustering of genes C – N.

Iteration	Two most similar objects		Correlation	New Object
	Object 1	Object 2		
1	L	G	1.00	[LG]
2	E	C	0.96	[EC]
3	N	H	0.95	[NH]

4	M	[NH]	0.95	[MNH]
5	[LG]	D	0.94	[LGD]
6	[EC]	[LGD]	0.94	[ECLGD]
7	I	F	0.60	[IF]
8	J	[ECLGD]	0.29	[JECLGD]
9	K	[JECLGD]	0.19	[KJECLGD]
10	[KJECLGD]	[IF]	-0.12	[KJECLGDIF]
11	[MNH]	[KJECLGDIF]	-0.96	[MNHKJECLGDIF]

Alternative Methods

k-means

The *k-means* method groups the genes into k clusters according to the genes' expression similarity. Initially, all the genes are distributed randomly into k groups. Each group may contain a different number of genes. Assume that each row lists expression levels at different time points for each gene, and each column contains expression levels of the different genes at one given time point. The *k-means* algorithm proceeds as follows. First, the average expression level for each column is calculated to create k pseudo genes, one representing each group. These pseudo genes function as seeds to start rearranging all the genes into groups. At this stage, each and every gene moves to a seed with the closest similarity, based on the chosen distance measure. This procedure iterates until there is no movement of genes. The process results in k clusters that contain genes with similar expression patterns. The value for k is chosen by the investigator.

QT Cluster

The QT Cluster algorithm finds large groups of genes that satisfy quality criteria. QT Cluster initially forms candidate clusters for each and every gene. Starting with the first gene, the algorithm finds the gene with the most similar expression profile to the starting gene, and adds this gene to the first candidate cluster. Now, we have two genes with very similar expression profiles in the first candidate cluster. To find the next gene to add to this group, QT Cluster uses the complete linkage method. That is, for each gene not in the candidate cluster, it computes the maximum distance to the genes that are in the cluster. The gene with the smallest maximum distance is added to the candidate cluster next. The algorithm stops adding genes to the first candidate cluster when no more genes with distance below the user-specified threshold can be found.

The same procedure is applied to the second gene to form the second candidate cluster. It is critical to note that the genes in the first candidate cluster are not eliminated from consideration when forming the second candidate cluster. Because a candidate cluster is formed from each and every gene, the number of candidate clusters is equal to the number of genes. And because the genes in the precedent candidate cluster are not eliminated from consideration when forming new candidate clusters, many candidate clusters overlap.

Once all the candidate clusters are formed, the candidate cluster containing the greatest number of genes labeled Cluster 1. Genes in Cluster 1 are removed from further consideration, and the entire procedure is applied to the smaller set of the genes to find Cluster 2. The algorithm stops when the number of genes in the remaining largest cluster is less than the user-specified minimum cluster size.

Data Presentations

By now, you may have forgotten why we were doing all this math. However, this is a good example of why biology students need to know math and ideally some computer science as well. There is a big need for biologists that can do math and computer science, and an even bigger need for computer scientists who understand some biology. But, back to our story.....

The purpose of all this number crunching was to identify which genes have similar behaviors under the experimental condition being investigated. The Tables 3 and 4 contain the answers, but they are not in a user friendly format. Humans are visual beings, and we need to see better displays of the information. There are two common ways to display these data. The simplest way is to generate a dendrogram, or a “tree picture”. A dendrogram uses a branching pattern to show the degree of relatedness among several objects. Two objects with a high degree of similarity are placed next to each other with a very short branch length (figure 5). Some dendrograms provide a scale bar so you can calculate the distance between two objects or leaves (keeping the tree theme going). The branch points mark the separation in clustering steps (see Table 4) used to put two objects together. You can learn more about dendrograms at this URL www.bio.davidson.edu/courses/genomics/seq/treeparts.html. If you want to generate a dendrogram using amino acid sequences for the enzyme IDH, you can go to this URL www.bio.davidson.edu/courses/genomics/tree/tree.html. MAGIC Tool allows you to choose among 4 different clustering methods.

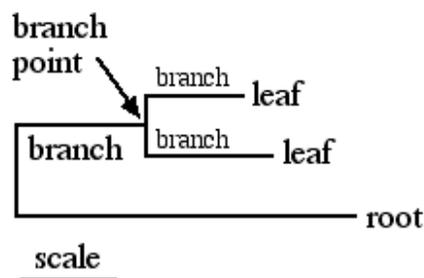


Figure 5. Dendrogram terminology.

Using a dendrogram, we can visually display Table 4 in a more meaningful manner (figure 6). In this dendrogram, you can see that the most closely related two genes are G and L as indicated by the branch terminating between these two leaves because the correlation of 1.0. From that point, you can see the shortest branch is the one that links genes E and C, followed by the branch linking M, N and H. This pattern continues as you gradually look for longer and longer branches indicating lower and lower similarities. It is important to note that the vertical order of genes M through F is not important. H could have been on top just as easily as M. So be sure to look at the

branching pattern and not neighboring genes. For example, genes H and K are adjacent in the dendrogram, but they are very dissimilar to each other as indicated by the long branches separating them.

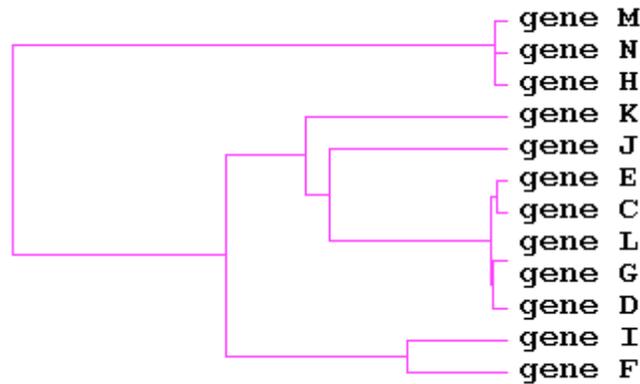


Figure 6. Dendrogram depicting the relationships of genes in Table 4.

Dendrograms are one way to display relatedness of genes, but they do not tell you which genes were induced and which genes were repressed. For this reason, many people modify the dendrogram by showing a color-coded representation of gene activity (figure 7). If you always think of green as the control cDNA and in the denominator of the ratio, then the color code makes sense. Those genes that were repressed are coded green and the greater the repression, the brighter the color green. Conversely, induced genes are colored red and bright red indicates induced genes. Black indicates genes that were equally transcribed under both conditions (i.e. a ratio of 1). Remember that ratios do not indicate the amount of mRNA present, just the relative differences in mRNA produced. Therefore, black does not indicate zero transcription; black indicates equivalent transcription.

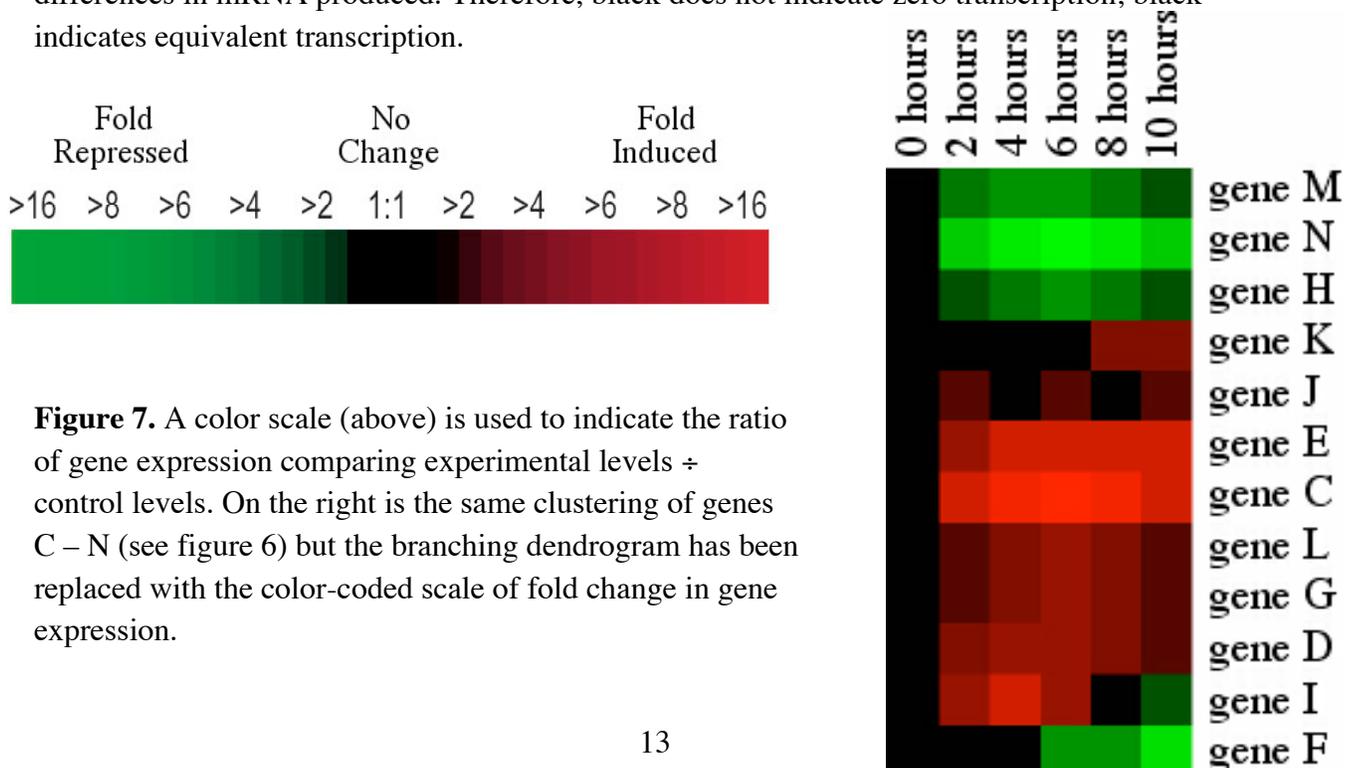


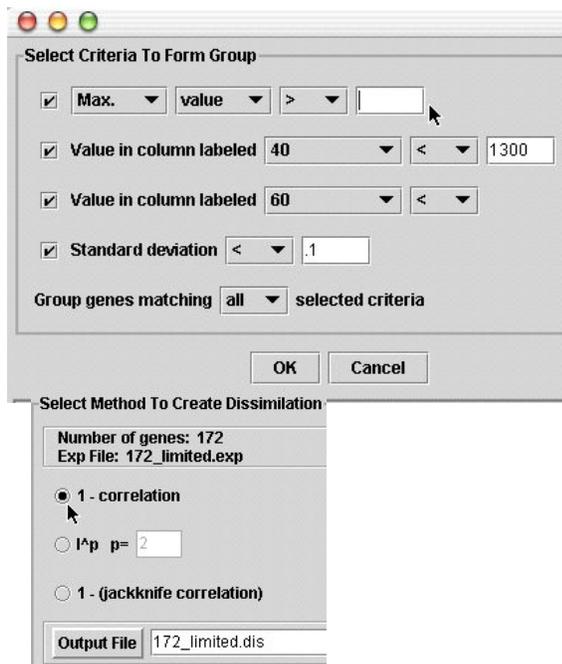
Figure 7. A color scale (above) is used to indicate the ratio of gene expression comparing experimental levels ÷ control levels. On the right is the same clustering of genes C – N (see figure 6) but the branching dendrogram has been replaced with the color-coded scale of fold change in gene expression.

Most datasets with microarrays are thousands of times more complex than the one we have studied. However, the principles are the same. As you read more examples, you will see different color schemes and the boxes of color will usually be compressed vertically until they are thin lines. However, you can still detect patterns of gene expression and visually detect clusters of genes that reveal biologically relevant patterns.

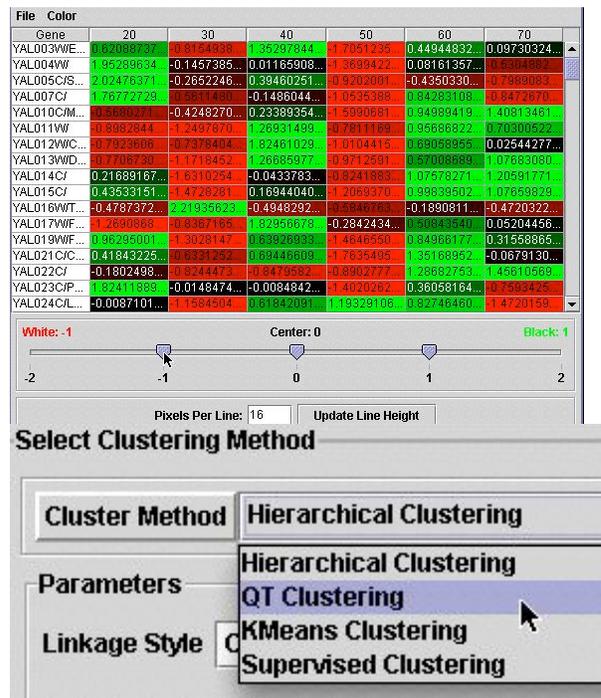
Exploring Data

Although the dendrograms are visually informative, you may want to explore your data rather than have the software do the exploration for you. For this purpose, you will need software that allows you to isolate genes based on characteristics you choose. One feature of MAGIC Tool is that you can create a hypothetical gene and have MAGIC locate any genes with similar profiles. As you explore, you can display the output in several ways. All of these tools are designed to let the user uncover interesting patterns that were undetected by the default settings on standard software algorithms. A montage of images are presented below to illustrate the options available through MAGIC Tool.

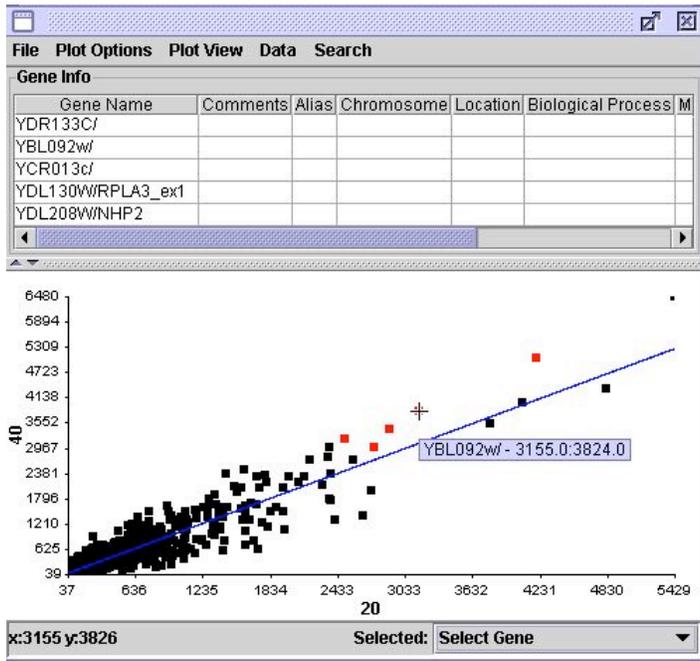
Choosing Search Criteria



Exploring ratios



Comparing two time points



Choosing display as graph

