SANDIA REPORT

SAND2003-4664
Unlimited Release
Printed December 2003

High Throughput Instruments, Methods, and Informatics for Systems Biology

George S. Davidson, David M. Haaland, Shawn Martin, Jerilyn A. Timlin, Michael B. Sinclair, Mark H. Van Benthem, Michael R. Keenan, Edward V. Thomas, Kevin W. Boyack, Brian N. Wylie, Jim Cowie, Juanita Martinez, Anthony Aragon, Margaret Werner-Washburne, Mónica Mosquera-Caro, Cheryl Willman

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Issued by Sandia National Laboratories, operated for the United States

Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865)576-8401
Facsimile: (865)576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.doe.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800)553-6847
Facsimile: (703)605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2003-4664
Unlimited Release
Printed December 2003

High Throughput Instruments, Methods, and Informatics for Systems Biology

George S. Davidson, Shawn Martin, Kevin W. Boyack, Brian N. Wylie
Computation, Computers, Information and Mathematics

David M. Haaland, Jerilyn A. Timlin, Mark H. Van Benthem, Michael R. Keenan
Chemical & Biological Sensing, Imaging and Analysis

Michael B. Sinclair
Microsystem Materials Tribology and Technologies

Edward V. Thomas
Independent Surveillance Assessment and Statistics

Sandia National Laboratories
P.O. Box 5800, MS-0318
Albuquerque, NM 87185-0318

Jim Cowie
The Computing Research Laboratory
New Mexico State University
Las Cruces, NM 88003-8001

Juanita Martinez, Anthony Aragon, Margaret Werner-Washburne
Department of Biology
University of New Mexico
Albuquerque, NM 87131

Mónica Mosquera-Caro, Cheryl Willman
Cancer Research and Treatment Center
Department of Pathology
University of New Mexico
Albuquerque, NM 87131

Abstract

High throughput instruments and analysis techniques are required in order to make good use of the genomic sequences that have recently become available for many species, including humans. These instruments and methods must work with tens of thousands of genes simultaneously, and must be able to identify the small subsets of those genes that are implicated in the observed phenotypes, or, for instance, in responses to therapies. Microarrays represent one such high throughput method, which continue to find increasingly broad application. This project has improved microarray technology in several important areas. First, we developed the hyperspectral scanner, which has discovered and diagnosed numerous flaws in techniques broadly employed by microarray researchers. Second, we used a series of statistically designed experiments to identify and correct errors in our microarray data to dramatically improve the accuracy, precision, and repeatability of the microarray gene expression data. Third, our research developed new informatics techniques to identify genes with significantly different expression levels. Finally, natural language processing techniques were applied to improve our ability to make use of online literature annotating the important genes. In combination, this research has improved the reliability and precision of laboratory methods and instruments, while also enabling substantially faster analysis and discovery.

Acknowledgements

The authors of this Sandia National Laboratories Report gratefully acknowledge the immense contributions of our collaborators, without which this work could not have been accomplished. Here we report the required Final Report for the Laboratory Directed Research and Development project High-throughput instruments, methods, and informatics for systems biology. However, much of the work to be discussed springs from our collaborators; we thank them for their generous time and patience in teaching us enough biology to be helpful, and for including us in their laboratories. Across the three years of this project we have worked directly with, and have had fruitful discussions with dozens of researchers from these laboratories and have benefited by many generous introductions to other researchers whom we would not have otherwise known. We owe each of these people our thanks for their critiques of our work, and encouragements to continue. Our collaborations with the University of New Mexico have been very fruitful. The informatics work reported here stems directly from close interactions with Vickie Peck who opened the world of genomics to us. Certainly, without the continuing encouragements and contributions of Maggie Werner-Washburne we would not have created useful tools, nor would we have been able to explain them to life scientists. One of the most important of these was Stuart Kim from Stanford University, who was able to see through the primitive nature of our early efforts and recognize what they could become. Our collaboration with Stuart stretched and improved all of our tools, and led to an important, joint publication that continues to have impact. The microarray work with Cheryl Willman's laboratory at the University of New Mexico Cancer Center drove the development of many of the statistical techniques presented here. Cheryl included us in her weekly laboratory meetings, which must surely have been made more tedious by our presence and the continuing need to teach us the rudiments of leukemia biology. Each of these groups is large and we have learned something from every one of you, for which we say thank you. We would especially like to acknowledge the help and collaborations of Moni Kiraly, Jim Lund, Kyle Duke, Min Jiang, Joshua M. Stuart, and Andreas Eizinger from the Kim Laboratory, and, of course, Edwina Fuge, Jose Weber, Juanita Martinez, Anthony Aragon, and Angela Rodriguez from the Werner-Washburne Laboratory. From the Willman Laboratory, we must certainly acknowledge Susan Atlas, Paul Helman, Robert Veroff, Erik Andries, Kerem Ar, Yuexian Xu, Huining Kang, Xuefei Wang, Fred Schultz, Maurice Murphy, and particularly Mónica Mosquera-Caro, and Jeffrey Potter who have been immensely helpful to our research. We would particularly like to thank Jon C. Helton, for suggesting the use of Savage scoring as a means to normalize microarray data. Leaving the world of biology and bio-medicine, our collaborations with Sergei Nirenburg, Jim Cowie, Chris Fields, and Valeriy Sibirtsev from the Computer Research Laboratory, CRL, at New Mexico State University have been fascinating. This collaboration has convinced us that Natural Language Processing (NLP) tools may become the most important application of computing in the entire analysis process for microarrays. If we have unfortunately omitted someone to whom we own our thanks and gratitude, please accept our apology, and recognize that we do value your help. As always, any misrepresentation, or error with respect to our collaborators' work is purely our own fault. Finally, we would like to thank the W. M. Keck foundation for funding the W. M. Keck Genomics Center at UNM, which was particularly important in our work, especially the in the development and construction of the hyperspectral scanner. The authors wish to acknowledge Michael R. Keenan for improvements in the MCR algorithms and Gary Jones for his aid in constructing the hyperspectral scanner. We thank Mary Anne Nelson for providing *Neurospora crassa* oligonucleotides for printing, and Gabriel Quiñones for technical support. This work was supported by grants from NSF (MCB-0092374) to M.W.W., an NSF Minority Post-doctoral fellowship to M.J.M, and USDA (99-38422-8034) to A. D. A. Development of the hyperspectral scanner was funded in part by the WM Keck foundation and a Laboratory Directed Research and Development program from Sandia National Laboratories. A portion of this work was also funded by the US Department of Energy Genomes to Life program (www.doegenomestolife.org) under project, Carbon Sequestration in *Synechococcus Sp.*: From Molecular Machines to Hierarchical Modeling," (www.genomes-to-life.org). Sandia is a multi-program laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-ACO4-94AL85000.

Introduction

The analysis of a complex system within an environment that is only subject to incomplete control is nearly impossible without some way to measure a large fraction of the system's internal state information. As a result, it is only with the recent advent of high throughput measurement technologies able to simultaneously measure tens of thousands of molecular concentrations that systems biology is really a possibility. As an example of the scope of this problem, consider that eukaryotic cells typically have on the order of ten thousand genes, each of which is likely to have several alternative splicing variants coding for the protein building blocks of the cell. These proteins undergo post-translational modifications and have multiple phosphorylations such that there are likely to be hundreds of thousands, or perhaps as many as a million variants. Hence the future of systems biology relies critically on high throughput instruments, such as microarrays and dual mass spectrometers. The research reported here addresses three important issues for such high throughput measurements: improved instrumentation for making the measurements, better methods to improve the precision of the measurements, and to avoid confounding main effects with process artifacts, and finally improved informatics to deal with the large volume of information from these techniques.

In the first section we present a new hyperspectral scanner, which is able to measure the complete spectra for each pixel, a major advance over available commercial scanners that can only measure light intensity through (typically two to four) filters. Importantly, this instrument has discovered major problems due to previously unrecognized contaminations that are often introduced in the manufacturing of the microarrays. Beyond being an important diagnostic tool, this hyperspectral scanner offers the potential to simultaneously measure many experimental conditions by using more than two fluorophores at a time, which could greatly increase the sensitivity of the experiments by controlling the "between array" errors.

Direct measurement errors are only one way that precision is lost. Experimental designs and continuous process control methods are essential to making the very best measurements possible for these very expensive experiments. Section two discusses these issues and presents particular results and findings.

Of course, the goal of better methods and instruments is to enable deeper understanding of the biology. The analysis tools and informatics systems developed to discover meaning in collections of these large-scale experiments are presented in section three. Here we address the structure of the typical data, its normalization, and ways to find important relationships.

Throughout each section we will present results and examples from our research to motivate the specific approaches, algorithms and analysis methods we have developed. We begin with hyperspectral scanner and microarray contamination issues, an important discovery enabled by this new instrument.

Section 1. Hyperspectral imaging of microarrays

Microarray technology is a relatively recent experimental development that allows high-throughput analysis of relative gene expressions of thousands of genes of an organism. The full details of the microarray process can be found in Schena.[1] In the standard microarray experiment, single-strand DNA gene fragments of known sequence are printed on glass slides in small spots on 150 to 250 μm centers. Up to 20,000 gene fragments (gene probes) can be printed on each glass slide. The microarray technology generally makes binary comparisons of gene expression from an organism for each microarray slide. The binary comparisons can be between cells in two different states or conditions such as between normal and abnormal (e.g., normal vs. cancerous cells). Messenger RNA (mRNA) is generated when a gene is being expressed in the cell, and the mRNA is subsequently extracted from the cells in the two states to be compared during the microarray experiment. The amount of mRNA is assumed to be proportional to the extent of gene expression of the cells in the two states. The mRNA from each cell type is then translated into single-strand cDNA (gene targets) and each labeled with a different fluorescent tag during the translation. The two labeled cDNA solutions are allowed to hybridize to the printed DNA attached to the microarray slide. The labeled hybridized microarray slide is scanned with one of several available commercial microarray scanners that are very sensitive optical filter fluorescence imaging systems. Specialized software is used to quantify the emission signal from the fluorescent label in the spot and the background signal around the spot. Ratios of the background-corrected and normalized signals yield quantitative measures of which genes are enhanced and which are repressed in the test sample relative to the control sample.

Microarray experiments have been demonstrated to be very effective for exploring the relative gene expressions of organisms under various conditions. Results from microarray experiments can be used to comprehensively and systematically explore the genome,[2] to identify genes involved in diseases[3], and to identify genetic predictors of treatment outcomes for cancer cells.[4] Although the microarray experiments have been extremely useful in expanding our knowledge of gene expression, microarray experiments can still benefit from improvements in the technology. For example, studies have demonstrated that the repeatability of microarray experiments within a microarray is much better than the reproducibility of the data between microarrays.[5] This variability limits the reliability of microarray experiments and as a result differences in gene expression less than a factor of two are generally not currently considered significant. Unfortunately, differences in gene expression for genes of interest are often expected to result in expression differences of less than a factor of two. In addition, the low expressed genes are often those that are of greatest interest, but the accuracy of measuring the low expressed genes is known to be less than that of highly expressed genes. Another significant limitation of current microarray experiments is that they are typically performed as binary experiments that limit the comparison of cells to only two states for each microarray slide studied.

Commercial microarray scanners generate separate high-spatial resolution images of each color filter channel on the microarray slide (which corresponds to each of the fluorescent labels).[6] All the currently available commercial microarray scanners use a separate laser or filtered white light source to excite each fluorophore tag. In addition, the emission of each fluorescent label is

separately monitored with the use of a single optical filter with the filtered light impinging on a photomultiplier tube or a CCD array detector. Because of the univariate nature of the current commercial scanner for each detected fluorescent label, the selection of dyes available for use is limited to those that have widely wavelength-separated absorption and emission spectra. The final signal obtained from the microarray for each spot involves the measurement and subtraction of the background emission (often measured from the emission of pixels surrounding the spot) from the total signal of the spot. High spatial resolution (≤ 10 μm) and sophisticated spot finding software are both required to accurately separate the DNA spot emission from that of the background. The assumption that must be made for the background correction to be valid is that the background emission obtained from the slide off the spot is the same as the background emission under the spot. In addition, any fluorescence from the glass, impurities, contaminants, etc. whose emissions overlap with the selected fluorescent labels will cause errors in determining the quantitative ratios of gene expressions if these sources of emission are not properly accounted for in the background correction procedure. We have demonstrated large quantitative errors in microarrays due to the presence of contaminant emission in the green channel of the scanner, and these results are the documented in a journal publication that was the result of this Laboratory Directed Research and Development (LDRD) project.[7]

Many of the limitations of the current commercial microarray scanners could be alleviated if the entire emission spectrum of each pixel were obtained and the spectral data quantified with appropriate multivariate analysis methods. Therefore, we have designed, built and characterized a new hyperspectral microarray scanner. There have been several previous reports of hyperspectral microarray scanners in the literature.[8-10] However, these scanners do not currently have the sensitivity of the commercial scanners, and because of the multivariate methods used to analyze the spectra from these scanners, they too are subject to quantitative errors if unexpected sources of emission are present in the data. The design of our new hyperspectral microarray scanner when coupled with analysis of the hyperspectral images with powerful multivariate curve resolution (MCR) analysis circumvents these limitations while maintaining the sensitivity level of current commercial scanners. It is the high-throughput, sensitive detection, and the use of MCR analysis that sets our system apart from others that have been developed. The MCR analysis in particular allows our system to yield reliable, accurate results even in those cases where unexpected sources of emission are present or where the spectrum of the dyes or glass substrate are different than expected. MCR analysis is a powerful multivariate method that enables us to perform quantitative spectroscopy on spectra without the use of standards. The implementation of MCR that we use here is a constrained alternating least squares analysis that iteratively solves for the pure emission spectra and the relative concentrations of each of the emitting components.[11-15] The concentration maps that can be outputted from the MCR analysis represent 2D images giving the spatial location and the relative concentration separately of each of the emitting sources. Emitting components can be separated and their concentrations accurately determined even when the spectra of the species emitting are highly overlapped spectrally and their locations on the slide are spatially coincident. Therefore, MCR is ideally suited for quantitative hyperspectral image analysis especially when some or all of the emitting species are not known. In the progression of this LDRD research, we developed new capabilities of MCR with the application of rigorous equality constraints that were required to achieve the results reported in this document. The details of these important new capabilities of MCR and example results have been documented in a recent journal paper.[21]

Our new scanner also offers the possibility of higher throughput microarray experiments by allowing many fluorescent labels to be used simultaneously on each slide. The use of multivariate analysis algorithms means that the fluorescent labels do not have to be widely separated in their absorption or emission characteristics. We will demonstrate that the new scanner has greater accuracy, higher sensitivity, and superior dynamic range than commercial scanners. The use of MCR analysis of the spectral data allows us to discover all emission sources on the microarray and to obtain relative concentration maps of each emission source whether the emission is from the fluorescent label, glass, or contaminants. The ability to obtain concentration maps of each emission species at each pixel means that separate background correction is not required with the new scanner. Because of these important features of the new scanner, it can be used not only for improved microarray analysis but also for improving the microarray technology by aiding the understanding of anomalous microarray data.

Of course, more accurate measurement of the microarray slides with the hyperspectral imaging system will not be an advantage if other sources of experimental variability dominate the biological signal that is being measured. Initial microarray experiments performed at the University of New Mexico demonstrated a significant lack of reproducibility even though published experimental protocols were carefully followed. Therefore, as part of this project, we developed experimental designs and performed numerous experiments to identify and correct experimental sources of variability that were present in our microarray data. Some of the results and conclusions from these designed experiments will be presented in this report.

Hyperspectral experiments

Hyperspectral Microarray Scanner Design and Operation

Full details of the design, operation, and characterization of the hyperspectral scanner have been submitted as a journal paper to Applied Optics: Optical Technology and Biomedical Optics.[16] Therefore, only a summary of the design, operation, and performance characteristics of the hyperspectral scanner will be given here. The new hyperspectral scanner is a “push-broom” design with line focusing of the excitation laser. Detailed discussion of the design of the scanner will be the subject of a later paper. A solid-state laser excitation at 532 nm was used for all the data presented in this paper. The laser light is focused to a line and reflected off a dichroic beam splitter into a 10X microscope objective to yield an excitation line with dimensions of 1 mm x 10 μm on the microarray slide. The emitted light from the microarray is passed through the microscope objective, through the dichroic beam splitter, and through a holographic notch filter to eliminate the laser emission. The filtered and focused line emission is imaged onto the slit of an imaging grating spectrometer. The line is dispersed by the imaging spectrometer onto a thermo-electrically cooled 2D CCD detector equipped with on chip electron multiplication gain. The electron multiplication gain serves to significantly enhance the signal relative to the read noise, and greatly improves the signal-to-noise ratio of the detector at low signals. The detector outputs a 16-bit digital signal. The current spectral range monitored by the system is 490 to 900 nm, but this range can be modified with changes in the spectrometer grating.

The microarray slide is mounted on an x-y positioning system. The second spatial dimension of the image is obtained by moving the slide under the microscope objective. Triggering signals from the positioners are sent to the camera to coordinate the collection of the emission signal with the position of the slide in 10 μm increments. Larger sections of the slide are imaged by stitching together successive 1 mm-wide scanned image sections.

The image data are corrected for the curvature of the imaged line on the CCD and calibrated for wavelength using the emission lines from low-pressure krypton and neon lamps collected through the optical axis of the scanner. Both lamps are necessary to obtain adequate signal across the spectral region of interest. The curvature is fit by least squares cubic polynomial fits of the peak locations of at least 15 emission lines from the lamps. A different cubic fit is found for each of the calibration lamp emission lines since the curvature varies with the y-position on the CCD camera and wavelength. Intensity variations along the length of the laser line projected on the sample are corrected for by normalizing the emission to the intensity of the laser line reflected from a clean glass slide. No attempts were made to calibrate the spectral emission to a radiometric source.

Many experiments were conducted using printed DNA microarrays and results from the hyperspectral scanner were compared to an Axon 4000B microarray scanner equipped with 532 and 633 nm laser excitation. The experimental procedure is captured in brief here. Microarray yeast gene expression slides were printed and hybridized with directly labeled Cy3 and Cy5 fluorescent dyes as described in Martinez, et al.[7] Commercially printed yeast microarrays and in-house printed microarrays (printed by our collaborators at UNM Department of Biology) were scanned by both the Axon and our hyperspectral microarray scanners to identify fluorescence from the glass, labeled DNA, and from contaminants if present. Each hybridized and labeled slide was scanned in the same region by both the Axon and hyperspectral microarray scanners. The 16-bit TIFF images of each channel from the Axon scanner were analyzed with the GenePix Pro software (Version 4.0 and Version 5.0) to identify spots and calculate spot intensities and ratios. Our multivariate analysis generated pure component concentration images as discussed in the future sections. These concentration images were exported from our software in 16-bit TIFF format and DNA spots were analyzed using the GenePix Pro software just as the commercial scanner images were.

In order to test the ability of the hyperspectral scanner and MCR analysis to quantify highly overlapped fluorophores, we printed Cy3 and Alexa 532 fluorophores on a microarray slide. The diluted pure dyes and 50/50 mixtures of the two dyes were printed along with two 10-fold serial dilutions of the pure dyes and the 50/50 mixture. This slide was then scanned with the hyperspectral scanner and the resulting spectra analyzed with our MCR software.

Multivariate data analysis

The MCR software was developed in house using Matlab Version 6.1 (The MathWorks, Inc. Natick, MA). Some of the capabilities of the software are described in more detail in the Theory Section. The MCR analysis was performed on Pentium 4 based personal computers that are 1

GHz or faster equipped with 1-2 GBytes of memory. The output of the MCR software includes pure-component emission spectra and relative concentration maps for each emission source. By always performing the MCR analysis on each slide, we were able to discover any unanticipated emission sources on the microarray and to observe unusual changes such as shifts in the expected fluorescence signals of the fluorophores. The MCR results presented in this paper are based on scans from just a portion of the slide (generally about 40,000 spectra). In these cases, the MCR algorithm generally converged after 20-100 iterations in a few minutes. Memory limitations originally restricted our analysis to these small spatial regions of the slides. However, recent enhancements to the MCR codes using principal component spectral and wavelet spatial compression and out-of-core-memory algorithms allow us to process and analyze spectral images obtained from the entire microarray with reasonable computation times that approach the time to simply read the data. Compression factors of nearly 200,000 have been achieved without loss of spectral or spatial resolution of the resulting pure-component spectra or 2D component concentration maps. After analysis of the hyperspectral images with the MCR software, the resulting 16-bit TIFF images of the concentration maps were imported to the GenePix software for further analysis.

Statistical Analysis of Microarray Slide Variation

Unfortunately microarray experiments are often dominated by variation other than the biology of interest that can mask the true gene expression relationships. In order to identify, quantify, understand, and correct sources of experimental variability in printed cDNA microarray experiments, a series of microarray repeat experiments were designed and performed at the University of New Mexico Biology Department. The microarrays were CMT S288C yeast v. 1.32 arrays (Corning) and were hybridized using the protocols described in Ref. 7. Only the signal intensities of the Cy3 dye were monitored due to the lack of Cy5 signal in the microarrays. The median responses of the Cy3 signal were analyzed without background correction since the background intensities were relatively constant and small in these data. Three replicate samples Sets A, B, and C were monitored from three different microarray specimens that corresponded to time course yeast samples taken at 10, 40, and 50 minutes. The experiments were designed to monitor and quantify the reproducibility and repeatability of operator, scanner, and hybridization on the Cy3 signal from the microarrays over short and long times. Two groups of data were obtained as follows:

Early data: Operator 1, Single replicate of set A; Operator 2, two replicates of set B; and Operator 3, two replicates of set C all taken within a period of one week. Later data were obtained one month later as follows: Operator 1, two replicates of sets {A, B, C}; Operator 2, single replicate of sets {A, B, C}; and Operator 3, two replicates of sets {A, B, C}. Short-term and long-term replicate-to-replicate variations were examined for a fixed operator and a fixed slide and between operators. In addition, operator-to-operator and slide-to-slide variations were examined. All the comparisons examined the variation separately for each of the 12 blocks located on the slides in order to investigate spatial effects on the slides. The replicated data were examined by fitting the two-way comparison data sets using robust linear regressions within blocks.

Theory

Multivariate curve resolution (MCR)

The MCR methods used in this work are based on constrained alternating least squares algorithms.[11] In all cases, rigorous least squares methods are used. The algorithms are based on classical least squares (CLS) calibration and prediction methods.[17, 18] In the following discussion, matrices are represented as bold uppercase letters, column vectors as bold lowercase letters, row vectors are represented as transposed column vectors, transposed matrices and vectors are denoted by a superscript T, and the pseudoinverse of a matrix is denoted by a superscript +. The CLS model for the fluorescence data assumes an additive linear model following the relationship,

$$\mathbf{D} = \mathbf{C}\mathbf{S}^T + \mathbf{E} \qquad (1)$$

where $\mathbf{D}$ is the $n \times p$ matrix of n spectra each containing p intensities as a function of wavelength, $\mathbf{C}$ is the $n \times m$ matrix of component concentrations where m corresponds to the number of components, $\mathbf{S}^T$ is the $m \times p$ matrix of pure-component spectra, and $\mathbf{E}$ is the $n \times p$ matrix of spectral residuals.

Principal component analysis (PCA) is generally used to determine the number of pure spectral species present in the data.[19] Often a semi-log plot of the singular values as a function of component number results in a clear demarcation to specify the number of components to include in the MCR analysis. We have found that if the wrong number of components is chosen, realistic pure emission spectra are not obtained. Once the number of pure emission spectra is chosen, we initiate the MCR algorithm with a guess for $\mathbf{S}^T$. Initial guesses for the pure spectra in $\mathbf{S}^T$ can be random numbers, principal component results, reasonable spectral shapes based upon known pure-component spectra, or pure-component spectra derived from previous MCR analysis of similar data. The CLS estimate for $\mathbf{C}$, denoted $\hat{\mathbf{C}}$, is obtained from

$$\hat{\mathbf{C}} = \mathbf{D}(\mathbf{S}^T)^+ \qquad (2)$$

Once $\hat{\mathbf{C}}$ has been obtained, the CLS estimate for $\mathbf{S}^T$ can be obtained from

$$\hat{\mathbf{S}}^T = \mathbf{C}^+\mathbf{D} \qquad (3)$$

However, there are infinite possible solutions to Eq. 2 and 3 due to rotational ambiguity of the solution. We can limit the range of possible solutions by employing constraints. Since the concentrations and pure-component spectra should be all nonnegative, we employ a nonnegatively constrained alternating least squares algorithm similar to that presented by Bro.[11] Improvements in the efficiency of Bro's algorithms have been implemented to dramatically reduce computation times. These improvements will be the subject of a future paper. The solutions to Eq. 2 and 3 are solved iteratively until the sum of squared spectral residuals converges to a specified tolerance level.

We also apply equality constraints when appropriate[20] to further limit the range of possible solutions to Eq. 2 and 3. The method of direct elimination is used when employing equality constraints in order to assure rigorous least squares solutions to the constrained problem.

Equality constraints can be applied when all or a portion of the pure-component spectrum and/or concentrations are known. They are also applied to compensate for a variable amount of offset signal present in our CCD detector output. In addition, if some components are known to be absent from a region of the image, an equality constraint with zero concentration can be applied to those pixels for components that are known to be absent. Our software possesses a great deal of flexibility when applying constraints to the alternating least squares algorithm. This allows us to use all of the spectral and spatial information known about a data cube and thus minimizes the rotational ambiguity converging toward a realistic solution.

Another improvement to the MCR algorithms that was developed in the course of related research in our group at Sandia National Laboratories involves weighting the data to accommodate the fact the noise in the fluorescence signal is dominated by counting or Poisson statistics. Poisson distributed noise has the characteristic that the variance of the noise is proportional to the signal. Optimal weighting of the data was implemented to make the noise distribution of the weighted data more nearly uniform. The optimal weighting is obtained by pre-multiplying the spectral data matrix by the inverse square root of the mean image and post-multiplying the data matrix by the inverse square root of the mean spectrum. MCR is applied directly to the weighted data. The resulting pure-component spectra and the component concentrations are then scaled by the corresponding inverses of the weighting matrices to obtain the results in the units of the original data.

Relating Hyperspectral Image Data to That of the Commercial Axon Scanner
After analysis of the hyperspectral microarray data with the MCR algorithms, the results can be quantitatively compared with the results from the image from the same microarray region scanned with the commercial Axon scanner. This comparison can be made since we know the optical filters used in the Axon scanner for both the Cy3 and Cy5 channels. After the MCR analysis of the hyperspectral microarray scanner spectra, we have the pure-component emission spectra for each emitting source. We can simulate Axon signals from the hyperspectral scanner data by digitally integrating the band pass of the optical filter with the concentration-weighted MCR generated pure-component emission spectrum. We can then separately generate the signals for each emitting source at each pixel from the hyperspectral scanner that correspond to the red and green signals that would have been measured if the optical filters had been present in the hyperspectral scanner. We sum these signals and force the sum to be equal to the corresponding Axon signal obtained using that filter. With this procedure, we can determine the portion of each emitting source that contributes to the two signals generated by the Axon scanner. We can also compare the fraction of fluorescent label that is present in each of the two signals from the Axon scanner.

Hyperspectral scanner results and discussion

Hyperspectral Scanner Identifies and Corrects for Contaminant Emissions

A freshly opened preprinted yeast microarray slide from Corning was scanned with the Axon 4000B scanner and processed with the Genepix software in a typical fashion (see Figure 1). This

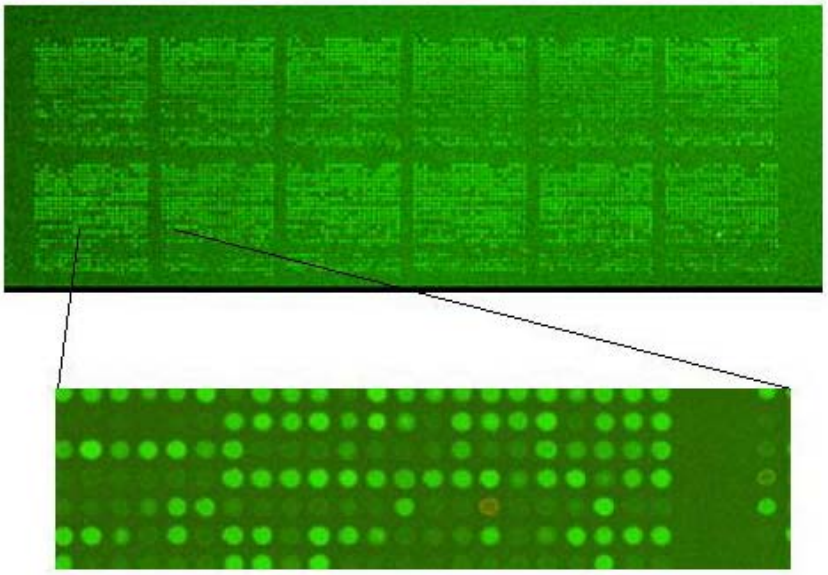


Figure 1. Red/Green ratio image of the recently opened, unlabeled Corning yeast microarray taken from the Axon microarray scanner. The expanded region of the slide corresponds to the area scanned with the hyperspectral scanner.

slide should not contain any emission from the DNA spots since it was not yet hybridized to the DNA with the fluorescent labels. However, it is clear from Figure 1 that the emission is greatest in the location of the printed spots. The quantitative ratio image indicates that this spot-localized emission is almost exclusively in the Cy3 (green) channel of the scanner. Similar spot-localized emission in the green channel was found for yeast microarrays obtained from three other commercial suppliers of yeast arrays and from our in-house printed yeast microarrays. Figure 2A shows representative emission spectra from the same slide scanned with the hyperspectral scanner using 532-nm laser excitation in the area of the inset in Figure 1. Figure 2B shows the pure-component emission spectra resulting from the application of our MCR analysis to the spectra in Fig. 2A. These MCR results were obtained using random positive numbers as starting points for the two pure-components identified as present by a PCA analysis of the spectra in Fig. 2A. Nonnegativity constraints were applied to the concentrations and pure-component spectra. The only equality constraint applied in this case was for the spectral offset present in the detector signal. Note that MCR generates only relative pure-component emission intensities and the pure emission spectra in Fig. 2B have been normalized to unit length. Comparison with published Cy3 and Cy5 spectra indicate that the emission spectra are not representative of either Cy3 or Cy5 emission.

Figure 3 shows the MCR generated concentration maps for both emitting species. The concentration maps make it clear that the source of one of the emitting species (the solid-line spectrum in Fig. 2B) is the glass since it is relatively uniform everywhere on the slide and its spectrum is similar to that of a clean glass slide monitored with our scanner. The other emitting component (the dashed-line spectrum in Fig. 2B) is the result of a spot localized contaminant that is introduced during the printing process. As we have found in other studies, this contaminant emission is a wide spread problem and only partially removed by standard hybridization and washing procedures.[7] The amount removed during hybridization and washing was found to be quite variable and unpredictable. Since most spots contain more contaminant emission than glass emission significant errors in the standard background correction will be present with current commercial scanners and software. These errors in background correction will result in significant quantitative errors in calculated Cy5/Cy3 ratios, especially when the measured Cy3 spot intensity is low to moderate.

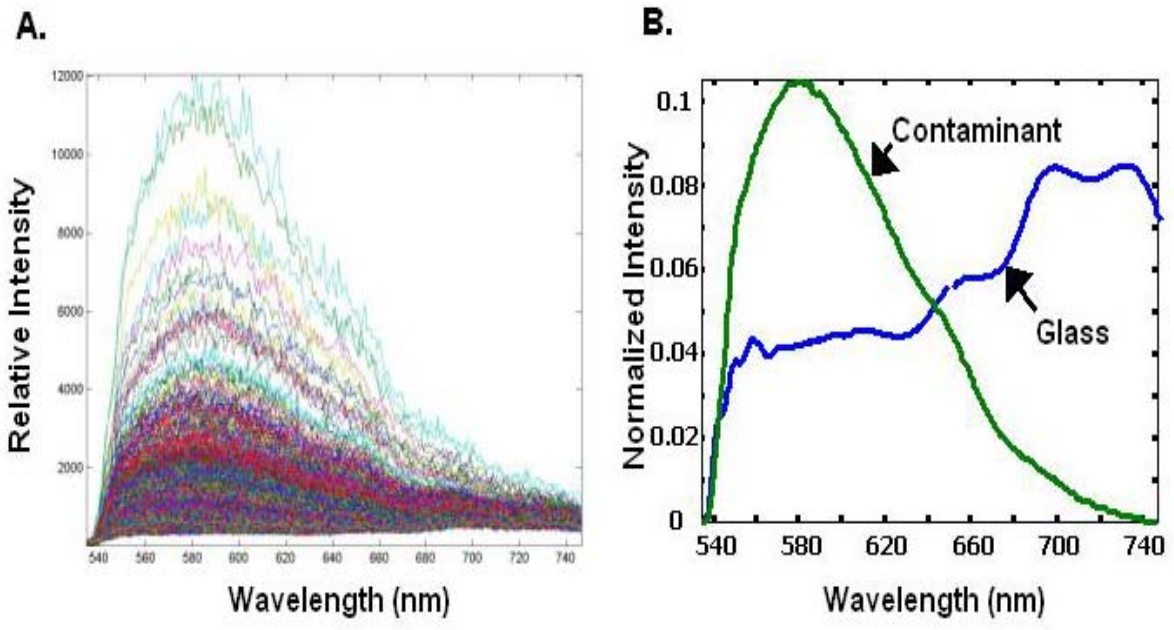


Figure 2. A. Emission spectra obtained from the expanded region of Figure 1 using the hyperspectral scanner and 532-nm laser excitation. B. MCR pure-component emission spectra of the glass and the contaminant extracted from the spectra in A.

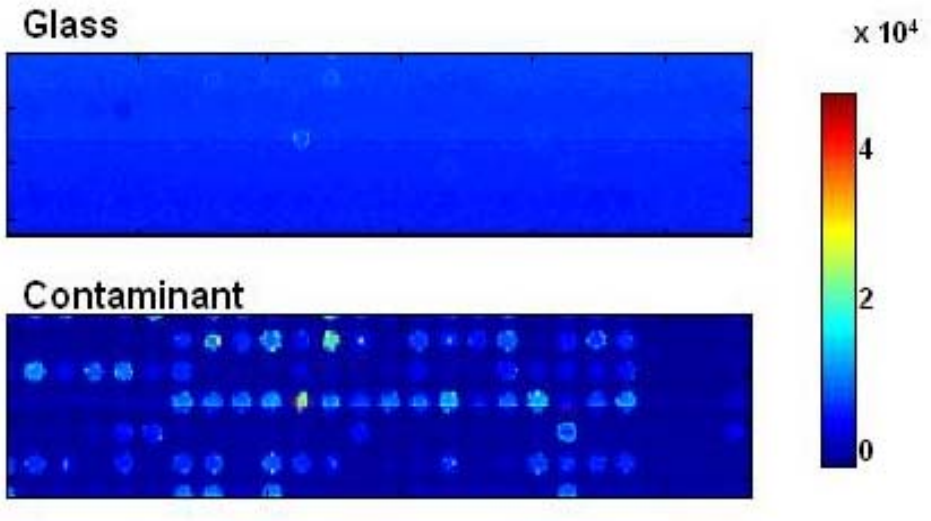


Figure 3. Relative concentration maps of the glass and contaminant emission species obtained by applying MCR to the analysis of the spectra in Figure 2A.

A yeast microarray from Corning was then hybridized with Cy3 and Cy5 labeled cDNA as described in Ref. 7. This slide was scanned by the Axon scanner and a portion of the slide was also scanned by the hyperspectral scanner with the 532-nm laser. Note that this laser simultaneously excites the glass, contaminant, Cy3, and Cy5 emission sources. because the Cy5 absorption is relatively low at 532 nm, its emission intensity is approximately a factor of 6 less than when excited by the 633-nm laser. Detailed results will be the subject of a future publication.[21] This publication will focus on the spectral components that would be confounded in the "green" channel of a commercial scanner. Figure 4 shows the spectra obtained from this microarray slide. PCA analysis of these spectra indicated that five pure spectral components were present. MCR analysis was performed on the spectra. However, in this case, the complexity of the spectra prevented the use of random numbers as starting points for the pure spectra in the MCR analysis. Instead, the glass and contaminant pure-component emission spectra were used as starting points along with Gaussian peaks generated to serve as close approximations to the Cy3 and Cy5 spectra (approximate width and wavelength positions of the dye emissions as determined from the literature). In addition, equality constraints were applied to portions of the pure emission spectra of Cy3 and Cy5 since spectral mixing of these dye emissions with the glass and contaminant emissions tended to occur at the extremes of the spectral range during the MCR analysis of these data. Therefore, the short wavelength portion of the Cy3 emission and the long wavelength portion of the Cy5 emission spectra were constrained to be zero consistent with the known spectra of these fluorescent labels. In addition, equality constraints were applied using a constant intensity spectrum normalized to unit length in order to fit the variable offset intensity present in the CCD output. Using these constraints along with non-negativity constraints for the emission spectra and their concentrations, excellent separation of the two dyes, glass, and contaminant emissions were obtained with the MCR analysis. Figure

5 shows the resulting glass, contaminant, and Cy3 pure-component emission spectra from the MCR analysis of the spectra obtained from this microarray.

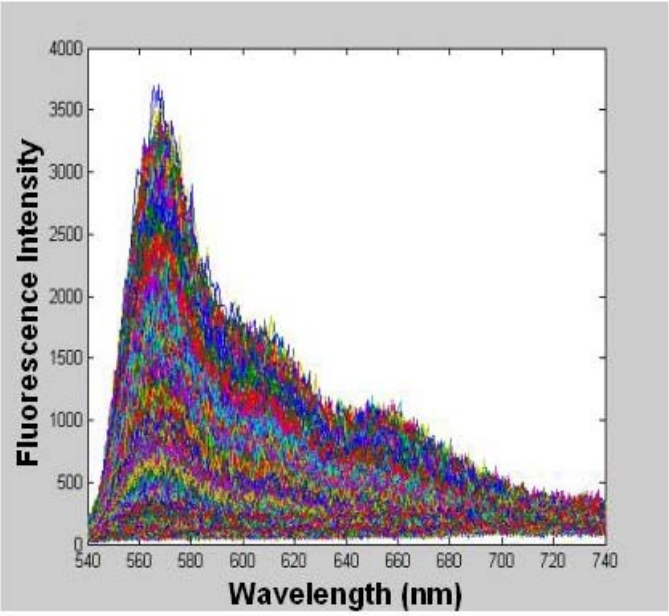


Figure 4. Spectra obtained from the microarray slide with Cy3, Cy5, glass, and contaminant emissions using the hyperspectral scanner.

Figure 5 also shows the green channel intensity map from the Axon scanner and relative concentration maps for the glass, contaminant, and Cy3 emission sources from the MCR analysis of the hyperspectral scans from the same region of the microarray. The total relative intensities for the MCR concentration maps were appropriately scaled as discussed in Section 3 so that the sum of the emission intensities in Fig. 5 corresponds to the total emission intensity for the Axon scan. In this manner, the total intensities for the glass, contaminant, and Cy3 measured by the hyperspectral scanner correspond to the total intensity measured in the Axon Cy3 channel. It is clear from Figure 5 that the glass emission is quite uniform, but the spot-localized contaminant is more intense than the Cy3 emission in most spots. The presence of this contaminant cannot be removed from the Axon images with the standard background correction methods since the contamination is spot localized. Thus, the error in the Axon (or any commercial microarray scanner) ratio images will be quite large for most spots in this slide. However, the ratios measured from the MCR analyzed hyperspectral scanner images will not have this error since the emission at each pixel is separately determined for all components. Therefore, neither the Cy3 nor Cy5 concentration maps obtained from the hyperspectral scanner are confounded by the presence of the glass or the contaminant emission.

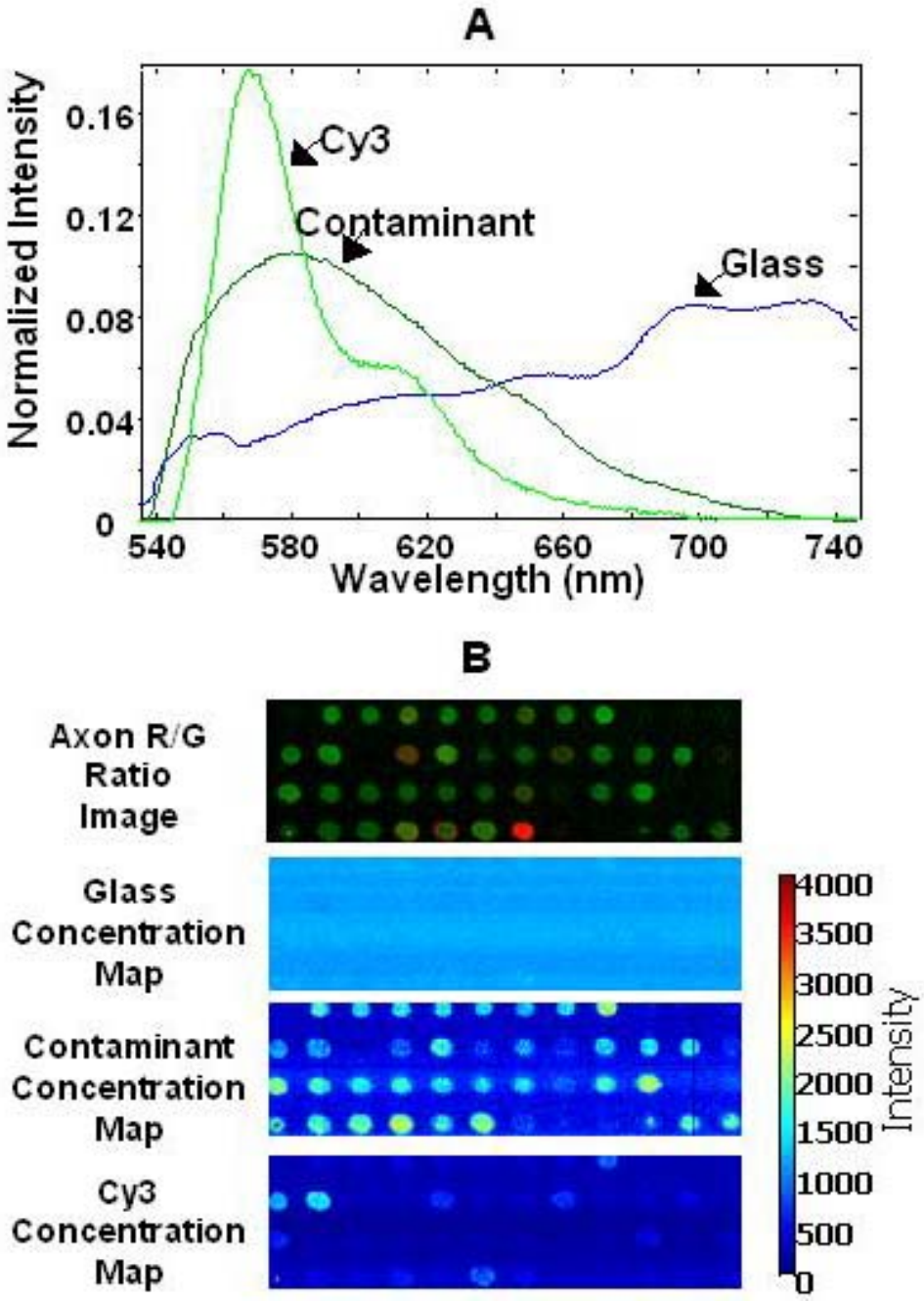


Figure 5. A. Pure-emission spectra of Cy3, glass, and contaminant obtained from MCR analysis of the hyperspectral scanner data presented in Figure 4. B. R/G ratio from the Axon scanner and the MCR generated concentration maps of the same area of the slide extracted from the hyperspectral scanner spectra.

If we pass the Cy3 and Cy5 concentration maps as TIFF files into the GenePix software, we can compare the Axon generated ratio images with the correct Cy5/Cy3 ratios obtained from the hyperspectral scanner using the same software to find and quantify the spot intensities for both dyes. The Axon generated ratio results would indicate a preponderance of green spots (Cy3 intensity > Cy5 intensity) whereas the ratio image based on concentration maps obtained from the hyperspectral scanner indicates that most of the spots have a higher Cy5 intensity than Cy3 intensity to yield a significant number of spots that are primarily red. From these data, a quantitative measure of the error in the Cy5/Cy3 ratio for this slide measured on the Axon scanner can be determined since the hyperspectral scanner yields ratios without the interference of the glass or the contaminant. As presented in Ref. 7, the calculated errors in the Axon-determined ratios indicate that approximately 75% of the spots are in error by a factor of 2 or more, 50% of the spots are in error by of factor of 3 or more, and 25% of the spots are in error by a factor of 4.5 or more. A boxplot illustrating these errors in the Cy5/Cy3 ratio is shown in Figure 6. The greatest errors are for the spots with low amounts of Cy3. Spots with high intensities in the Cy3 channel will be much less affected by the presence of the contaminant emission, and the ratios of these spots will have relatively low errors. Since this slide has low Cy3 emission intensities for most of the spots, the size of the errors obtained when using commercial scanners is relatively large. In addition, errors in the normalization of the "green" and "red" channel data on the commercial scanners will be in error if contaminating fluorescence is present in the microarray. We estimated that the error due to incorrect normalization for the above slide amounted to an additional factor of 2.2 based on the data from the commercial scanner. MCR analyzed data from the hyperspectral scanner are immune to these additional normalization errors.

In addition to providing higher accuracy for the Cy5/Cy3 spot ratios, the hyperspectral scanner is very useful in determining the source of artifacts and problems with a given microarray slide. For example, it is reported in the literature[22] that dye separation is apparent in spots on some slides. The apparent dye separation is observed as spots with red rings around a green center. It is difficult to understand the driving force for this apparent dye separation. We also observe occasional spots with red rings around a green center when our slide is scanned with the Axon scanner. However, with the capability of the multivariate curve resolution of hyperspectral data to quantify the relative concentration of each emission source at each pixel, we can determine that the presence of spots with red rings around a green center is not indicative of dye separation. For example, in a spot with the greatest variation of red/green ratios across the spot, we find that the Cy5 concentration for the spot is relatively high, the Cy3 concentration is low, but the contaminant concentration is high. (Figure 7) The Cy5 and Cy3 are spatially

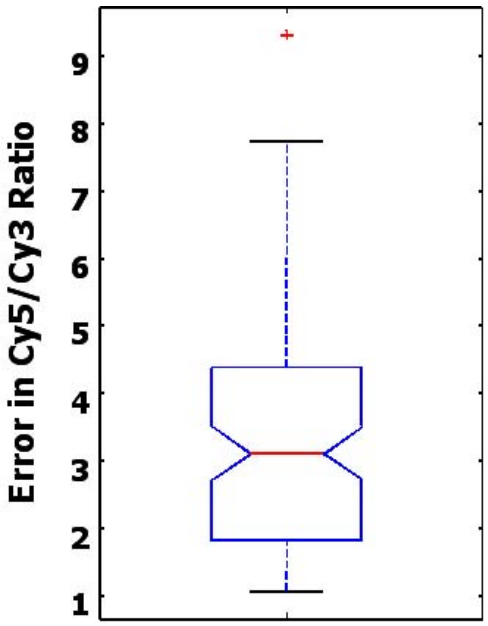


Figure 6. Boxplot illustrating error factor in Cy5/Cy3 ratio due to presence of green contaminant.

coincident in the spot, but the contaminant has a smaller diameter. Therefore, in our microarray slide, rather than dye separation, the appearance of a red ring around a green center is due to the fact that the contaminant is significantly more intense than the Cy3 and its size is smaller than that of the labeled DNA spot.

Another anomaly often described in the literature is the presence of negative spots or "black holes."[22, 23] In this case, the background around the spot appears higher than the emission under the spot. Since negative spot intensities are clearly not realistic, numerous papers have presented a variety of background correction methods such as using backgrounds obtained from portions of the slide far from the spots where the background intensity is low or using as background signal the intensity of control spots printed with DNA from another organism that is not expected to hybridize with the target cDNA to eliminate the presence of these negative spots. Although these alternative background correction methods minimize the presence of negative spots, they are subject to uncertainty since the background assumptions are more speculative and the source of the anomalous background can only be guessed. With the multivariate curve resolution of hyperspectral data, no separate background correction is necessary since the identity and relative concentration of each emitting source is determined at each pixel. Therefore, no assumptions about the spatial distribution of the background emission are required, and no subtractions of backgrounds away from the spots are required. Therefore, multivariate analysis of hyperspectral data can help researchers understand common microarray anomalies, and more details on the analysis of these anomalies will be presented in a future paper.[21]

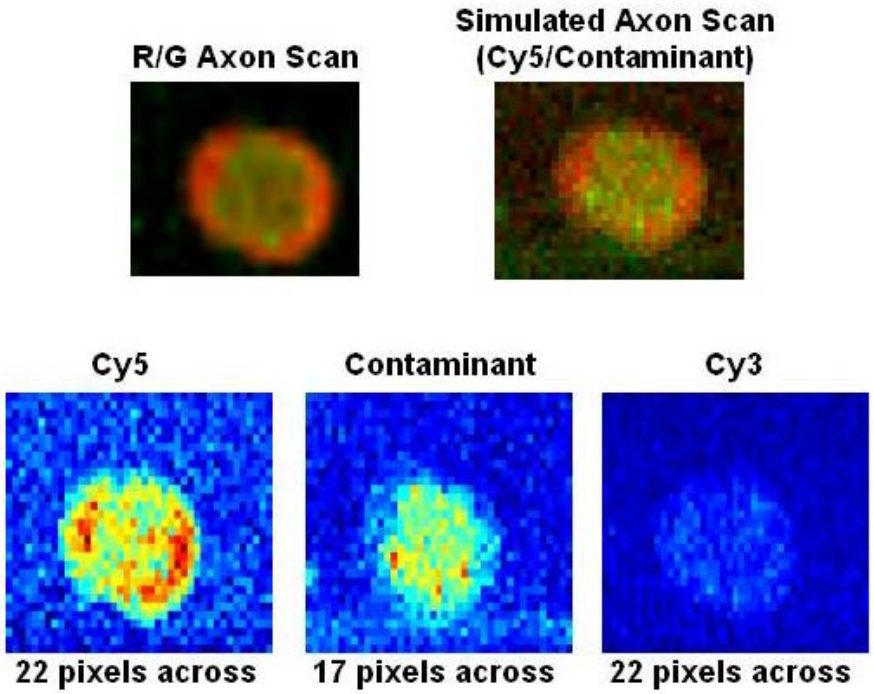


Figure 7. Illustration of apparent dye separation.

Finally, the higher throughput capabilities possible with the hyperspectral scanner have been tested using two closely overlapping dyes (Cy3 and Alexa 532) printed on a glass slide at various concentrations of pure and mixed dyes. The MCR analysis of the spectra obtained from this slide demonstrate the ability of the scanner to separate and quantify these two dyes even though their emission maxima are separated by only 12 nm. Figure 8 shows the separated concentration maps of the two dyes and the pure-emission spectra extracted for the glass, Cy3, and Alexa 532. These results clearly demonstrate the potential for higher throughput capabilities with gene expression microarrays with multiple overlapping dyes. Experiments are currently underway to demonstration of the ability of the hyperspectral scanner to dramatically increase microarray throughput in an actual microarray experiment with multiple overlapping dyes incorporated into the hybridized cDNA.

Hyperspectral scanner summary

We have demonstrated the power of our newly developed hyperspectral microarray scanner coupled with multivariate curve resolution to yield improved accuracy, better background correction, and a more complete understanding of microarray data. From our experiments on calibration slides with Cy3, we have been able to determine that our scanner has greater

sensitivity than the commercial Axon scanner.[16] Our scanner also has a higher dynamic range than the commercial scanners since the spectrum is widely dispersed over many detectors with 16 bit A-to-D converters. The multiplex advantage of the MCR analysis allows our scanner to make use of the entire measured emission spectrum whereas the commercial scanners are limited to the narrower wavelength range of photons passing the optical filter. The potential for the new scanner to improve the throughput of each microarray slide has been demonstrated by measuring the ability of the scanner to quantify highly overlapping dye emissions.

Currently we have been only scanning portions of the microarray slides with the hyperspectral scanners. The multiple gigabyte size of the full slide images has been a hindrance to analysis of these large files. However, data compression methods in both the spectral and spatial dimensions along with recent advances and improvements in the efficiency of the MCR algorithms make possible the rapid analysis of the full slide hyperspectral images on standard PCs. In addition, the fact that separate background corrections are not necessary with the hyperspectral scanner means that high spatial resolution of the current commercial scanners is not necessary. We have also shown that lower spatial resolution images can yield comparable quantitative results relative to the high spatial resolution images (i.e., 10 μm vs. 30 μm spatial resolution) that are generally collected by commercial scanners. Since comparable results have been demonstrated with lower resolution scans, the rate of data collection can be increased, the size of the files greatly decreased, and the speed of the MCR analysis further improved.

The combination of microarray scanning with a very sensitive, high-throughput hyperspectral imaging system and the accuracy and understanding of the data made possible with the MCR data analysis makes the scanner an important new tool for improving microarray technology. In addition, the imaging system is not restricted to scanning microarrays. For example, the hyperspectral imaging system might be able to quantitate GFP fluorescence in the presence of overlapping non-specific fluorescence from cells and growth media. The hyperspectral scanner makes possible the use of many more variants of GFP than possible with current confocal imaging systems. Our new hyperspectral imaging system and multivariate curve resolution can also monitor multiple fluorophores in stained or labeled tissue sections. We have also completed a design for a new prism-based imaging spectrometer that will improve optical throughput of the system by a theoretical factor of three and will decrease image curvature to less than a single pixel. A new backside-illuminated detector has been obtained that will increase the sensitivity by a factor of two, has twice the number of pixels, and 10 times the read speed of our current CCD detector. These improvements that are in progress will greatly improve the speed, sensitivity, and image quality of the current microarray scanner. In addition, new funding has provided us with the components and labor to add a third dimension to our hyperspectral scanner. Therefore, we will soon have the capability of 3D hyperspectral imaging at the diffraction limit. Thus, the future prospects for our new imaging system are bright.

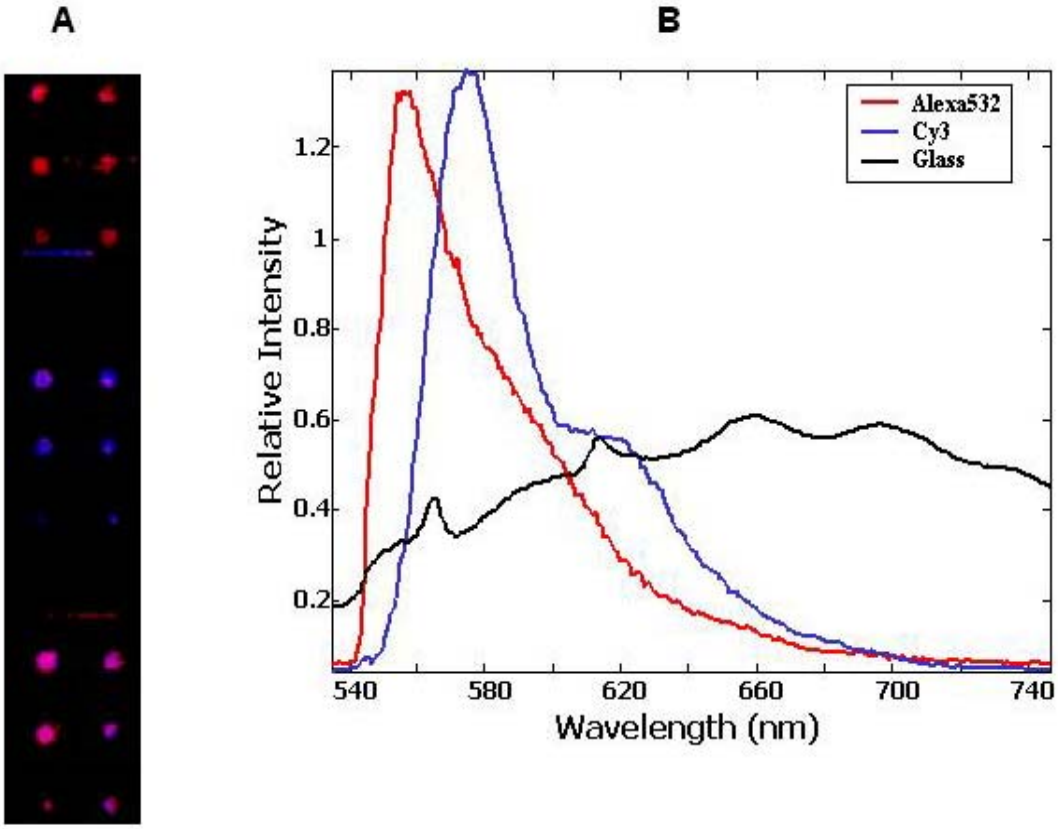


Figure 8. A. Composite image created from MCR-generated concentration maps of Cy3 and Alexa 532 fluorophores determined from MCR analysis of hyperspectral data collected from slide of spots printed with variable concentrations of pure dyes and 50/50 dye mixtures. B. Pure-emission spectra of Cy3, Alexa 532, and glass based on MCR analysis of hyperspectral scanner data from the slide presented in A.

Section 2. Designed experiments reduce errors

The results of the designed replicate microarray experiments demonstrate that Operators 1 and 3 exhibited good short-term repeatability while Operator 2 exhibited short-term repeatability that was dependent on the block examined. Further examination of Operator 2 analyses revealed that for one set of replicates the starting rows of four of the twelve blocks in one case and two of the twelve blocks in another case were incorrectly identified due to very weak spots present in the first rows of these blocks. The result was an analysis that was comparing different genes between replicates for the blocks with the misidentified rows. Therefore, a simple training procedure was adequate to correct this problem that might not have been uncovered without the replicate analyses. The analysis of the short-term operator-to-operator variability was found to be quite good if the blocks from the misidentified rows were eliminated from the comparison for

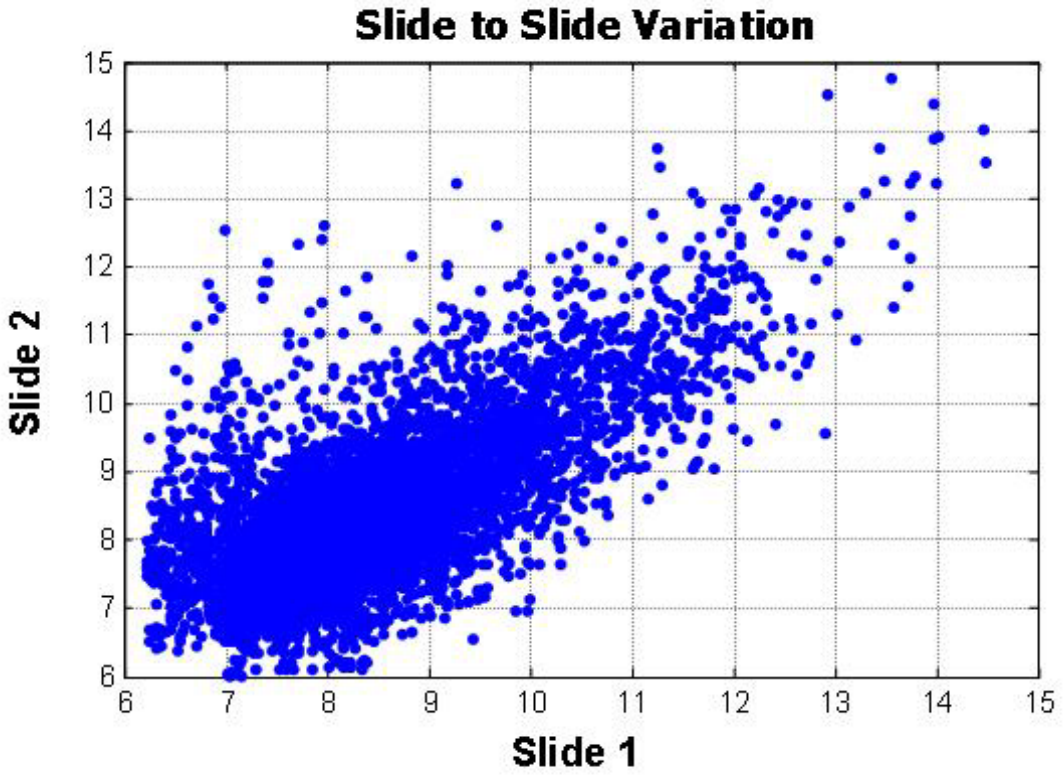


Figure 9. Slide to slide variation for two slides in statistical study of sources of variation in microarray experiments.

Operator 2. Similar high repeatability was also observed for the month long separation of the replicate scans. These results would indicate that the operator variability over time is small and that the scanner is both reproducible and very stable.

When the slide-to-slide variation was examined, we found that significant repeatability problems existed. Figure 9 demonstrates an example scatter plot for Cy3 for two replicate arrays measured by the same operator over a short time period. In Figure 9, the intensity levels for all spots in the array are shown. The results demonstrate that for a few genes, the measured expression levels vary by more than two orders of magnitude for these replicate hybridizations that started with the same mRNA samples. In order to understand the source of the repeatability problem, the data were analyzed by block. These results are shown in Figure 10 for the same slide represented in Figure 9. It is now observed that, in general, the data in the individual blocks are highly correlated but that the slopes vary by block and tend to decease systematically from the top left to the lower right-hand side of the array. This observation would indicate some lack of homogeneity in the hybridization that is most likely caused by lack of adequate flow of the solution over the array during the hybridization process.

In order to test the hypothesis that adequate mixing during hybridization was the source of the repeatability problem, a new set of designed experiments were performed after changing the hybridization process. Instead of using cover slips which require capillary flow for mixing of the hybridization solution, we used cover slips with lifters to facilitate better fluid flow. The use of the cover slips with lifters does require more hybridization solution to be used, but the mixing should be more complete. In addition, the microarrays were place on a mechanical rocker during the temperature-controlled overnight hybridization to further promote mixing of the hybridization solution. A representative plot of the results from this improved experiment is shown in Figure 11 for slide-to-slide variability. As observed in Figure 11, the between slide variability has been dramatically reduced with all but a small number of spots being reproduced to with a factor of two or less. Thus, the new hybridization procedure has successfully improved the slide-to-slide repeatability. Because of the significantly decreased variability, the second set of designed experiments was able to identify other sources of variability that were not apparent

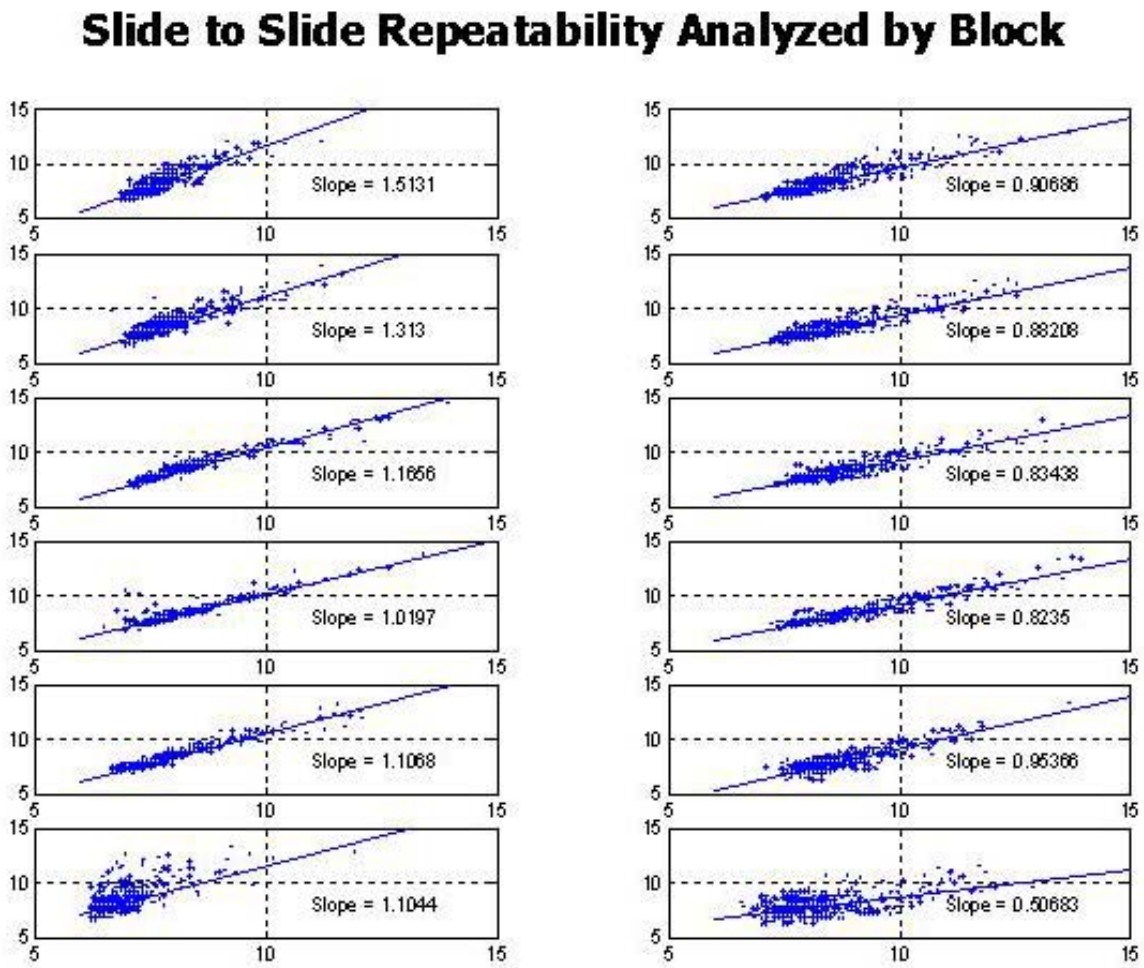


Figure 10. Slide to slide variation for two slides in statistical study of sources of variation in microarray experiments.

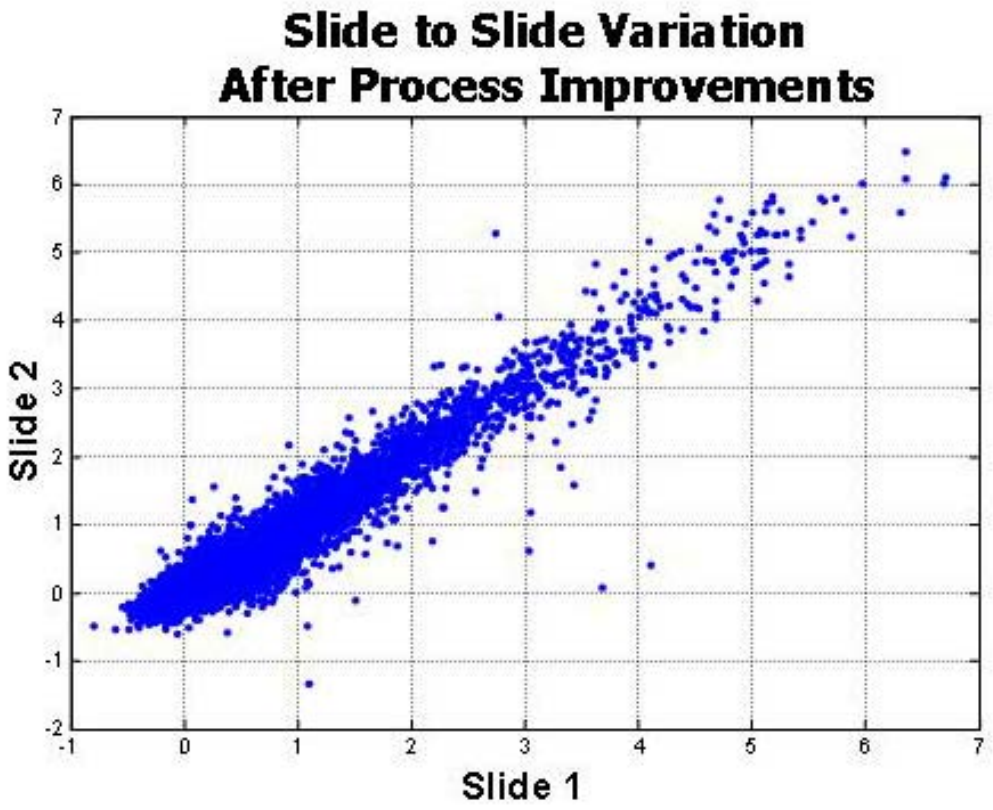


Figure 11. Slide to slide variation is much improved after improvements to major sources of variation in the microarray manufacturing process.

in the first experiment. The new experiments identified variability due to the printing process of the microarrays. Thus, we have initiated a process of continuous improvement for the microarray experiments.

Section 3. Informatics for microarrays

Informatics issues and introduction

Microarray experiments and their analyses seek to detect effects in gene expression under different treatments or natural conditions with the goal of clarifying the cellular mechanisms involved in the cells' differential responses. Uncertainty is the rule rather than the exception in

these analyses. First, the underlying systems (cells and/or tissues) are incredibly complex whether viewed from the dynamic process perspective, or their physical realizations in space and

time. Second, there is abundant variability between cells experiencing exactly the same conditions as a result of genetic polymorphisms, but also because of the stochastic nature of these chemical systems. Third, the collection and initial preservation of these cells is not a precisely controlled process. For example, when a surgeon removes a cancer the tissue may not be frozen or otherwise processed for minutes to hours, meanwhile the cells continue to respond to these unnatural conditions. Further, the tissues, or partially processed extracts, are often sent to another laboratory several hours or even days away from the original collection site, all of which offers opportunities for chemical changes. Fourth, these measurements are not easy to make; they involve many processing steps with a wide variety of chemicals, and at every step variability arises. The processing is often, necessarily, divided across several days and among several technicians, with inherently different skills and training. Further, the chemicals are never perfectly the same; they are created in batches and age, both of which affect the laboratory yields and the quality of the processing. Finally, the arrays themselves are technological objects subject to all sorts of variability in their creation, storage, and final use. In effect, the simple measurement of mRNA concentrations that we would like to make is confounded by huge uncertainties. To be able to make good measurements it is essential that all of the mentioned steps be subject to careful statistical process control monitoring and systematic improvement, and further, that the actual experiments be designed to avoid, randomize, or otherwise balance the confounding effects for the most important experimental measurements. These are *best practices* more often found in their breach than in actual laboratory experience, unfortunately.

By the time the data are ready for analysis they are typically presented in a numeric table recording a measurement for each gene across several microarrays. For example one might be analyzing 400 arrays each with 20,000 gene measurements, which would be presented in a table with 20,000 rows and 400 columns. Often the table will have missing values resulting from scratched arrays, poor hybridizations, or scanner miss-alignments, to name just a few from among the host of possible problems. The analysis methods should be able to gracefully deal with these incomplete datasets, while the analyst should approach these data with great skepticism and humility considering how complicated the cellular processes are, and how error prone our microarray technology is. Despite all of these issues, statisticians and informaticians, unlike mathematicians, are expected to say something about the structure and meaning of these data. Because, as Thompson has said, "[statisticians] *should be concerned with a reality beyond data-free formalism*. That is why statistics is not simply a subset of mathematics."[24]. Here, we attempt to follow Thompson in discussing implications, as well as our approach to analyzing these experiments, including considerable detail about the algorithms, and the way the data are handled.

In general, we begin by preprocessing the measurements with thresholding, rescaling, and various other transforms. We then compare the genes by computing pairwise similarities with several techniques. These similarities are used to cluster (or assign spatial coordinates to) the genes in ways that bring similar genes closer together. These clusters are then visualized with VxInsight.[25, 26]The clusters are then tested with statistical methods to identify genes and

groups of genes that are differentially expressed in the identified clusters, or genes otherwise identified with respect to experimental questions and hypotheses. The expression values for the identified genes are plotted, and tested for stability. Those genes which seem particularly diagnostic or differentiating are studied in detail by reading the available information in online databases and in the original literature. To help with this expensive and knowledge intensive literature review we have developed the text analysis tool, Genome Literature Exploration Environment (GLEE), which automatically gathers together, and restructures available textual information in ways that match our analysis protocol thereby enabling a smooth cognitive flow during the analysis. Each of these analysis steps will be presented in an order approximately following the analysis order we use in practice.

An overview of the basic clustering

Organizing large groups of data into clusters is a standard and powerful practice in exploratory data analysis. The first step after the initial data transformations involves the pairwise comparisons of the data elements to determine their relative similarity. This comparison, typically, results in a single similarity number. For example, when comparing the expressions of N genes across multiple experimental conditions one might compute N(N-1)/2 correlation coefficients as the similarity measure between each possible pair of genes. After the data pairs have been assigned similarities, various grouping algorithms can be used, for example hierarchical clustering (which produces dendrograms) or the force-directed clustering algorithms developed at Sandia National Laboratories. However, each of these approaches expects to find a single similarity value (the result of applying a single similarity criterion) for the data pairs. In the next section we show one way to prepare the raw data for similarity processing, then turn to computing the similarities, and consider the population structures of the pairwise similarities in typical microarray data.

Data transformations and similarity computations.

As discussed earlier microarray data typically have a large number of missing data, or values otherwise deemed to be non-present. We typically drop genes with too many missing values, where that threshold is under the control of the analyst. Then the raw values are scaled to help with the processing.

The distribution of microarray measurement values typically have extremely long tails, that is, there are a few genes with very large expressions, while most of the others are quite small. Figure 12 shows the distribution of raw counts from an Affymetrix U95AV2 chip, which exhibits the typical distribution, similar long-tailed distributions are found for other array technologies. The extreme differences between the larger values and the more typical values cause problems with most analysis methods. Tukey suggested a number of transforms to make data from such distributions less extreme and more like the normal Gaussian distribution[27]. In particular, taking logarithms of the raw data is a common practice to make microarray data more symmetric and to shorten the extreme tail, see Figure 13.

However, we frequently use another transform to compress the extreme values, which is due to Savage[28]. This rank order based score is an increasing function of expression level. Howerver, the smaller values are compressed to be very nearly the same, which is particularly useful with array data where a very large component of these smaller values is purely due to noise. If the expression levels are rank ordered from smallest to largest, $X_{(1)} \le X_{(2)} \le ... \le X_{(n)}$, then the score for $X(k)$ is given by $SS(X_{(k)}) = \sum_{j=1}^{k} \frac{1}{(n+1-j)}$.

Figure 14 shows how this score compresses the extreme tail, and Figure 15 compares the log transformed values with the SS values for an intermediate range of raw counts. Figure 16 show the distribution of savage scores for 8943 genes. About 60% of the savage scores are below 1, see the savage score percentiles in Figure 17.

The use of this scoring has two advantages over correlations using raw counts. First, because it is based on rank ordering, data from arrays processed with very different scalings can still be compared. Second, because the noisiest fraction of the measurements is aggressively forced toward zero, the effect of the noise is suppressed without completely ignoring the information (it has been taken into account during the sorting phase) in the genes with low expressions. Large differences in rank order will still be strong enough to be detected.

The normalization of array data is controversial. Some form of centering and variance normalization might be a good approach. However, it has been argued that for many experiments there is no intrinsic reason to expect the underlying mRNA concentrations to have the same mean (or median) values and variance adjustment is even more suspect[1]. Nevertheless, the analyst has the option to do such normalizations, if desired. In general, we avoid this issue by working with order statistics and savage scores. After adjusting the numbers to achieve the desired normalizations and distribution adjustments, the similarities must be computed as discussed in the following section.

[1] Personal correspondences with Stuart Kim.

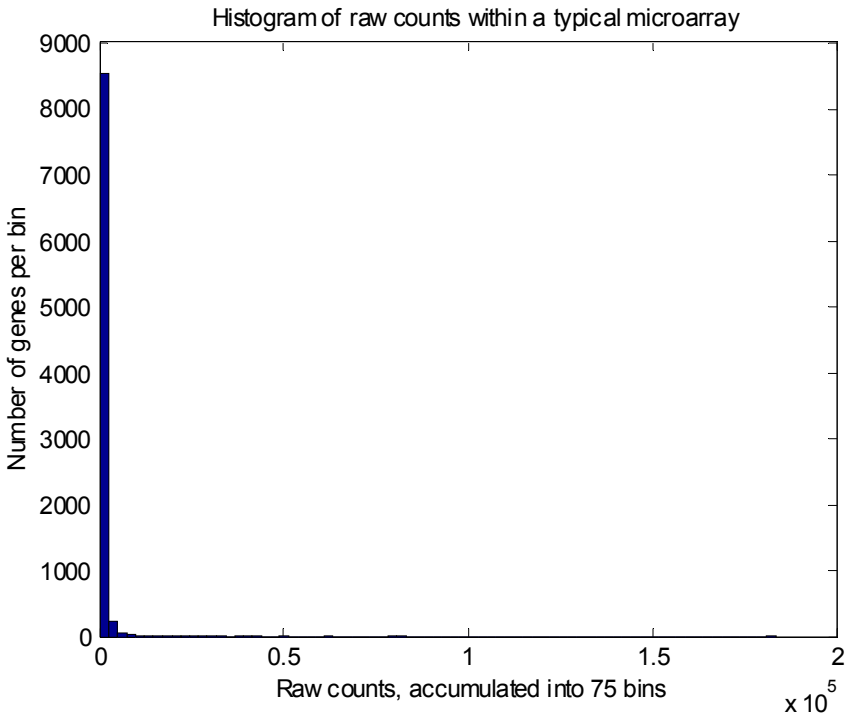


Figure 12. The distribution of expression levels in raw counts from a typical Affymetrix U94A microarray. Note the few extreme values greater than 50,000 counts, while most measurements are less than 5,000.

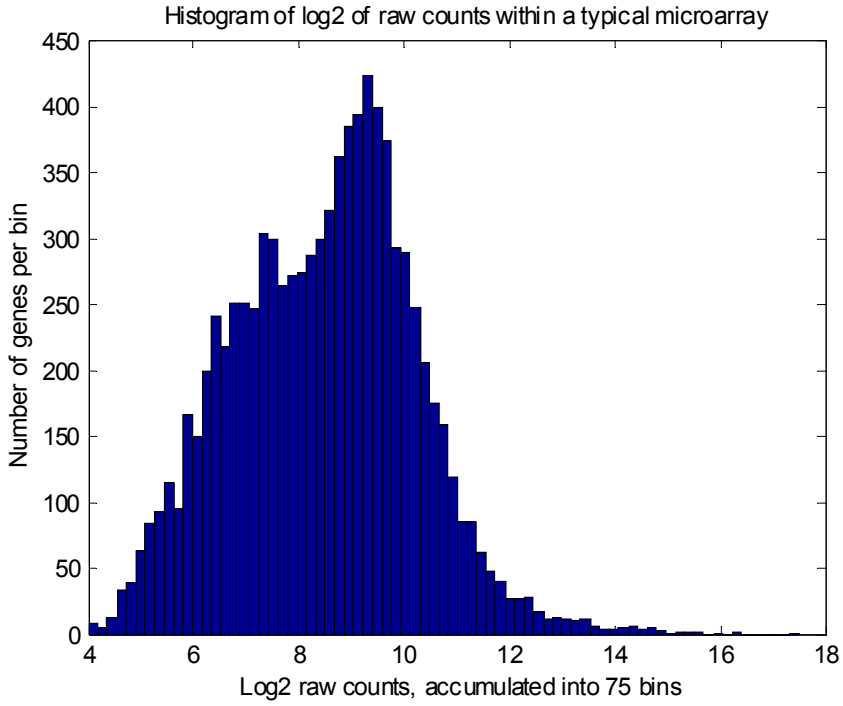


Figure 13. The distribution of expression levels from the previous figure, but now after log transformations.

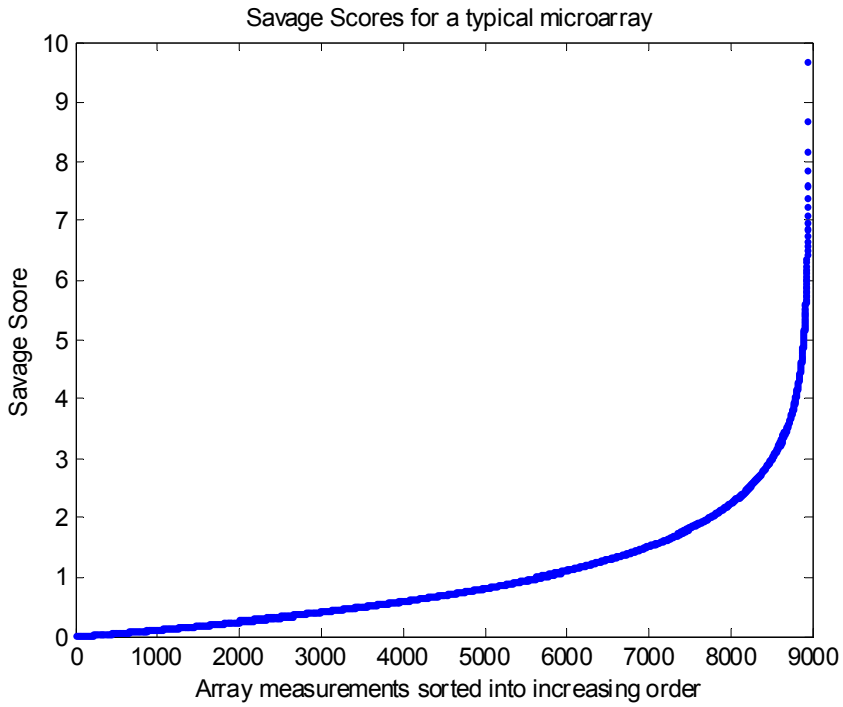


Figure 14. The expression levels from Figure 12 after compressing with savage scoring.

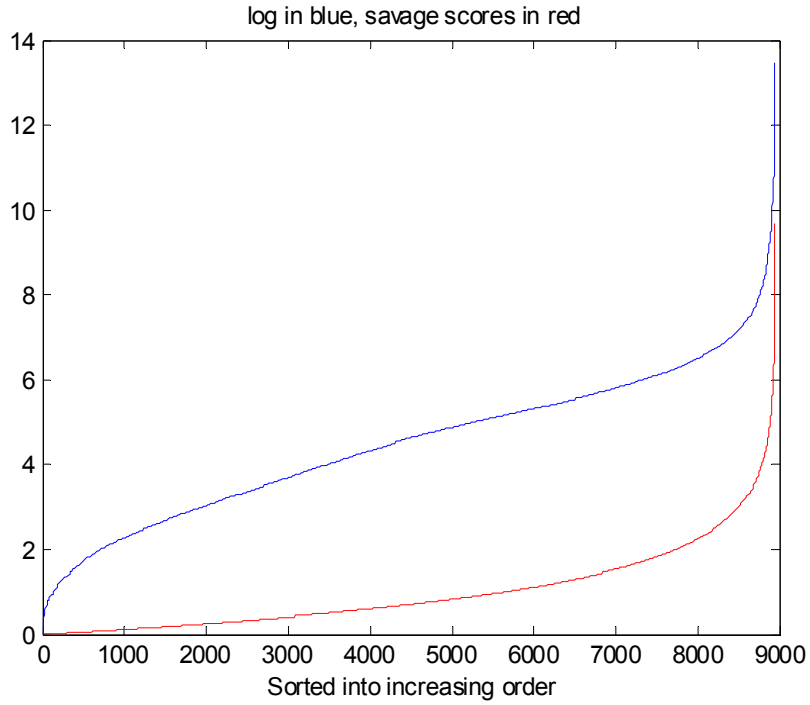


Figure 15. Notice how savage scoring is much more aggressive than log transformations in compresses the extreme values.

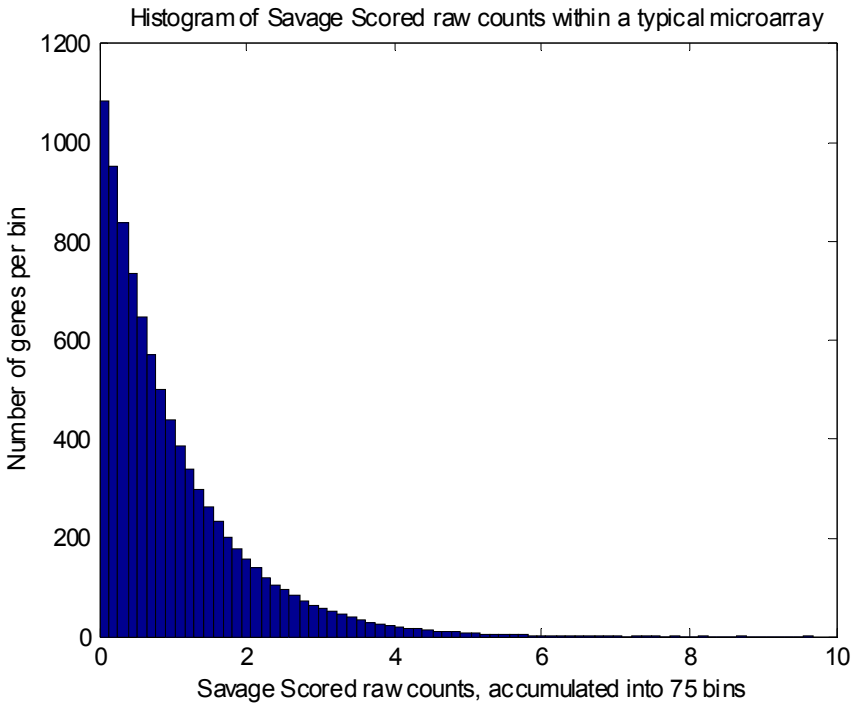


Figure 16. A histogram of savage score values for 8,943 genes. Note that, in contrast to log transformed expression levels, savage scores are bound in a predictable way.

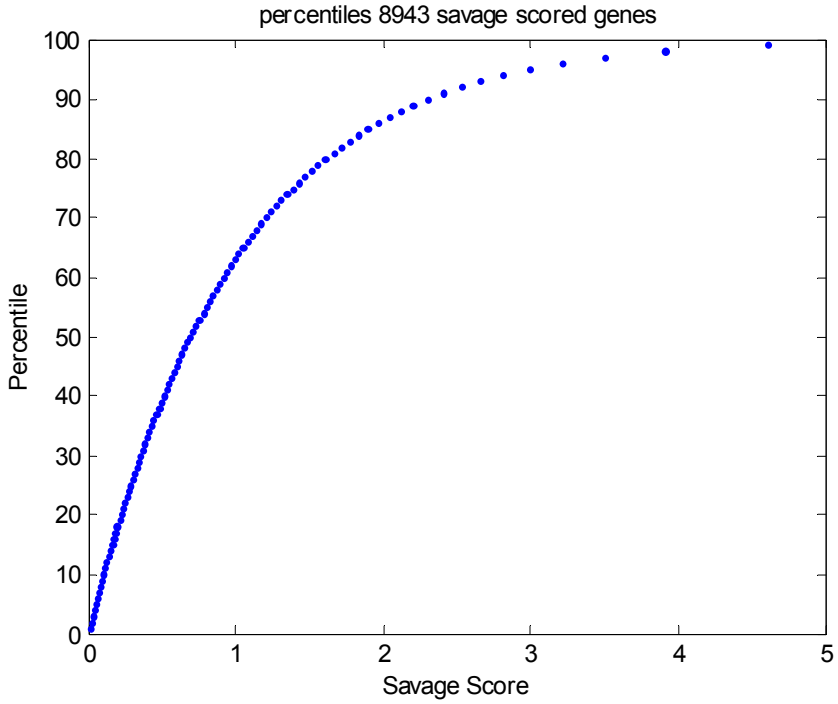


Figure 17. A percentile plot for the savage scores in the previous figure. Note that about 60% of the genes are below 1.

The data as an abstract graph

The relationship between the data elements and their similarity values can be visualized as an abstract, weighted graph $G(V_i, E_{i,j}, W_{i,j})$ consisting of a set of vertices, V, (the genes) and the set of edges, E, with weights (the similarities between the genes), as shown in Figure 18. This graph is only topologically defined; the vertices have not been assigned spatial locations. Spatial coordinates are computed from the weighted graph using the iterative clustering algorithm VxOrd,[26]which places vertices into clusters on a two dimensional plane such that the sum of two opposing forces is minimized. One of these forces is repulsive and pushes pairs of vertices away from each other as a function of the density of vertices in the local area. The other force pulls pairs of similar vertices together based on their degree of similarity. The clustering algorithm stops when these forces are in equilibrium. The clusters are then visualized with VxInsight[25, 26], which represents the clusters as a mountainous terrain built above the clustered vertices, which have been collocated to the extent that they are similar to other vertices in the local neighborhood. The height of each mountain is proportional to the number of vertices under it, and the separation between mountains is an indication of the degree of dissimilarity between the groups of vertices under one mountain with respect to the nodes under the other mountains. The details of this process will be discussed below, but one should note that the process has not specified any particular method for computing similarities, and, in fact, the entire process can be used with any real-valued similarity measure.

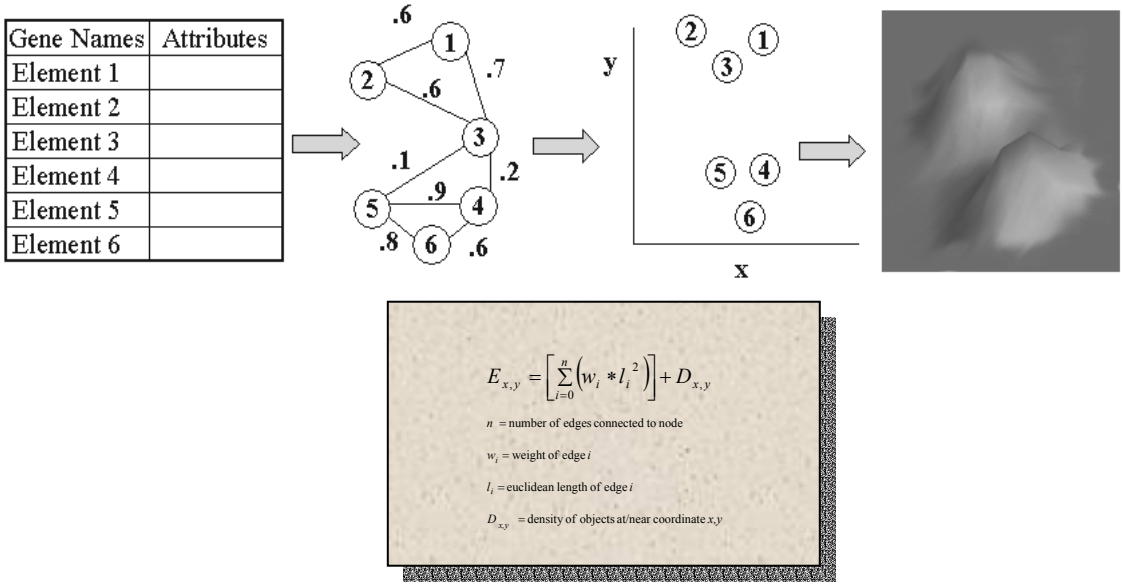


Figure 18. Data elements are nodes and similarities are arc values, which are clustered and assigned X,Y coordinates and represented as a mountain range by VxInsight.

Choosing a similarity measure

One obvious candidate for measuring similarities is the simple correlation coefficient, R, due to Pearson[29],

$$R_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}, \text{ where, of course, } -1 \le R_{xy} \le 1.$$

Pearson's R is just a dot product of the two n-dimensional vectors which have been mean centered and normalized by their lengths, so one can think of R as a measure of the extent to which the two vectors point in the same direction in the n-dimensional space. Of course the vectors might lie along the same line, but point in opposite directions, which is the meaning of $R_{xy} = -1$. If the vectors are completely orthogonal the correlation will be zero.

In fact, Pearson's correlation is the measure of similarity that we, and the rest of the microarray community, use most often. It is, however, not the only possibility, and in fact has some features that do not recommend it under certain situations. For instance, Pearson's correlation is known to be sensitive to outliers, especially when n is small. Technically, R, has a break-down point of 1/N, meaning that as few as one extreme outlier in the N points can make the statistic completely different from the true measure of correlation for two random, but correlated variables[30]. In practice, we have not found this to be a real problem with sets of arrays running into the hundreds. However, early in the development of microarray technology many data sets were published with order ten arrays. In these cases it was deemed valuable to apply more computationally expensive, but more robust measures of correlation. Details about robust measures of correlation, including the Percentage-Bend Correlation Coefficient can be found in[31].

We have also found occasion to use very different similarity measures. For example, we clustered genes based on the textual similarity of their annotations in the online Mendelian Inheritance In Man (OMIM) database [32, http://www.ncbi.nlm.nih.gov/omim/]. In this case, the text (see Figure 19 for a typical gene annotation) was processed with the RetrievalWare search and retrieval package from Convera. RetrievalWare computes the similarity between two text documents with a proprietary algorithm, based on word co-occurrences, and word importance by the location of the word's occurrence in the sentence. For each gene annotation the strongest 19 other annotations were accumulated to create a similarity file and then processed as previously shown in Figure 18. The gene clusters based on the textual similarities of their annotations in OMIM is shown in Figure 20, which is presented solely as an example of the use of alternate similarity measures within the overall process. With our present text analysis tools, the clusters do not correlate with clusters based on microarray measured gene expressions from 254 leukemia patients as shown in Figure 21. The more typical processing for microarrays is discussed in the following section.

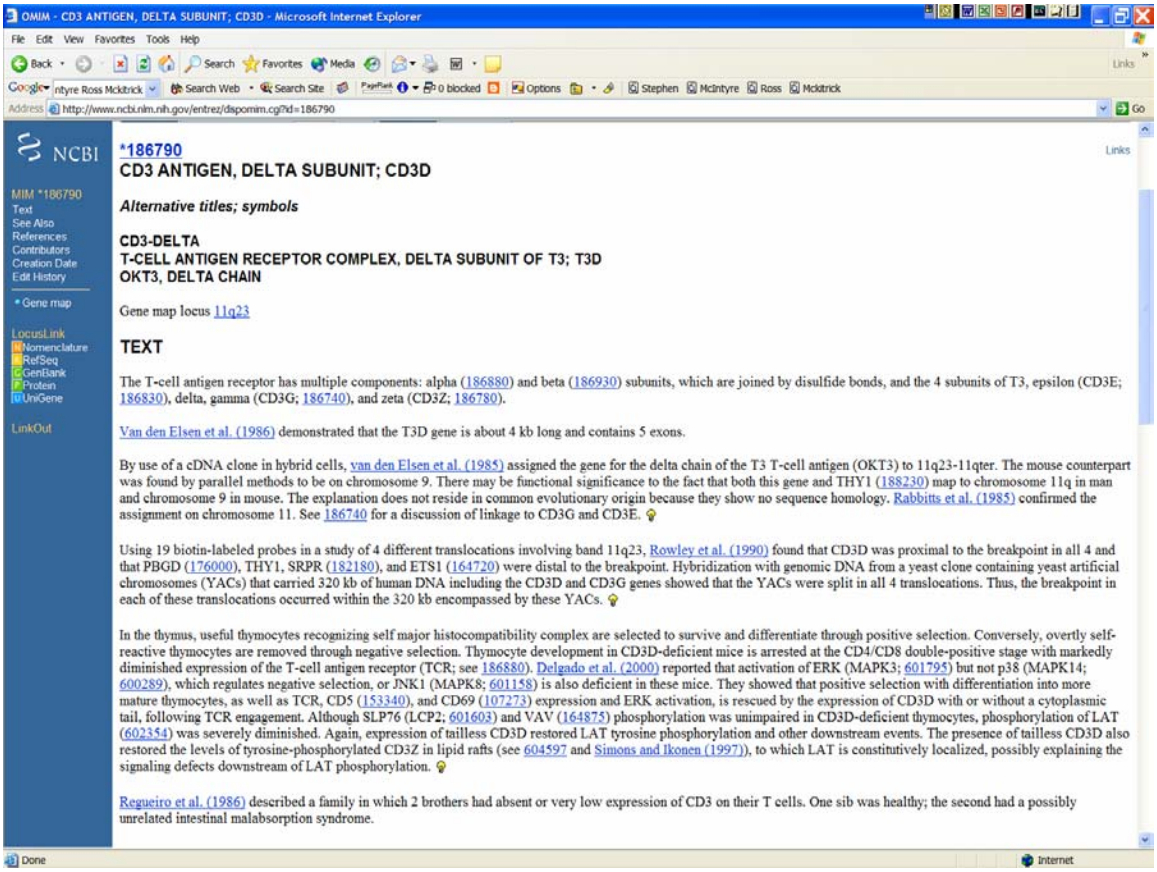


Figure 19. Typical text annotating the individual genes in the OMIM database.

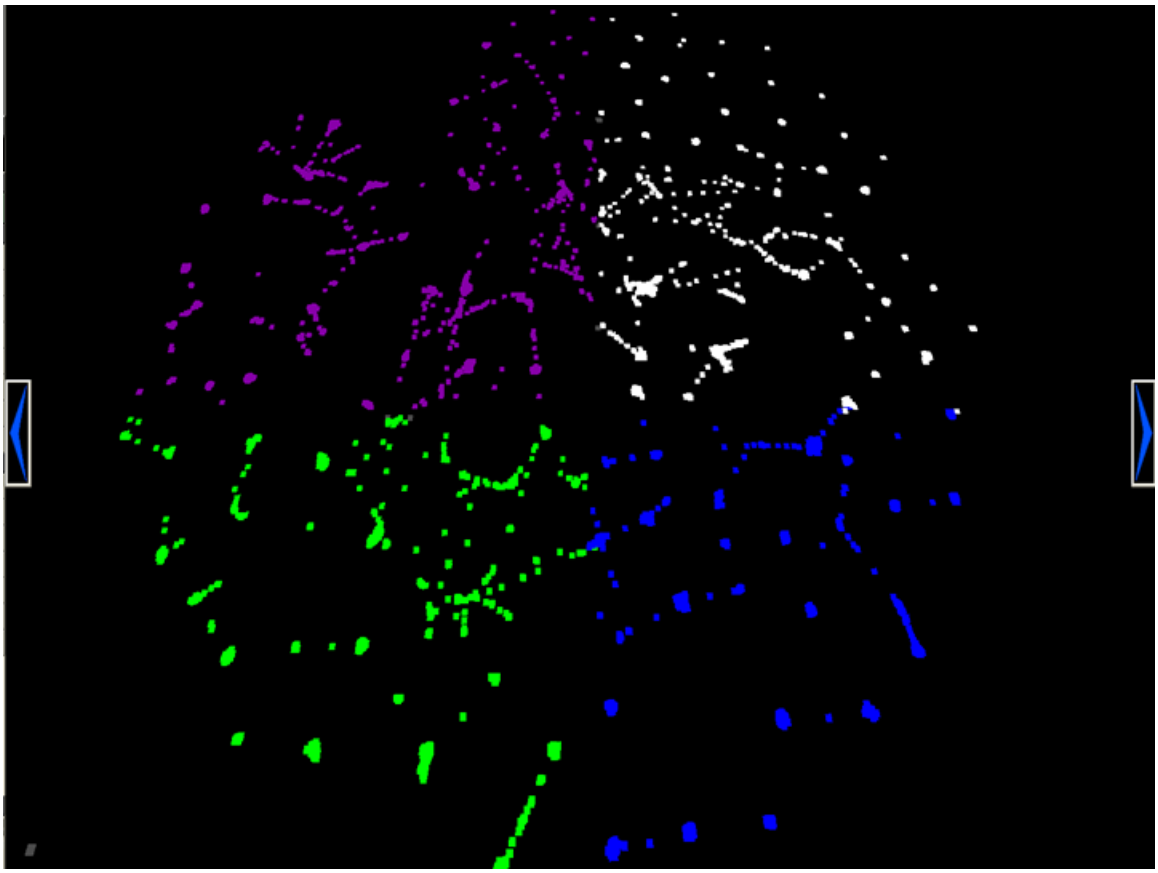


Figure 20. Cluster of genes in the OMIM database using similarities computed from the annotation text itself. Four quadrants have been colored here for comparison with the clusters in Figure 21, which are based on actual gene expressions.

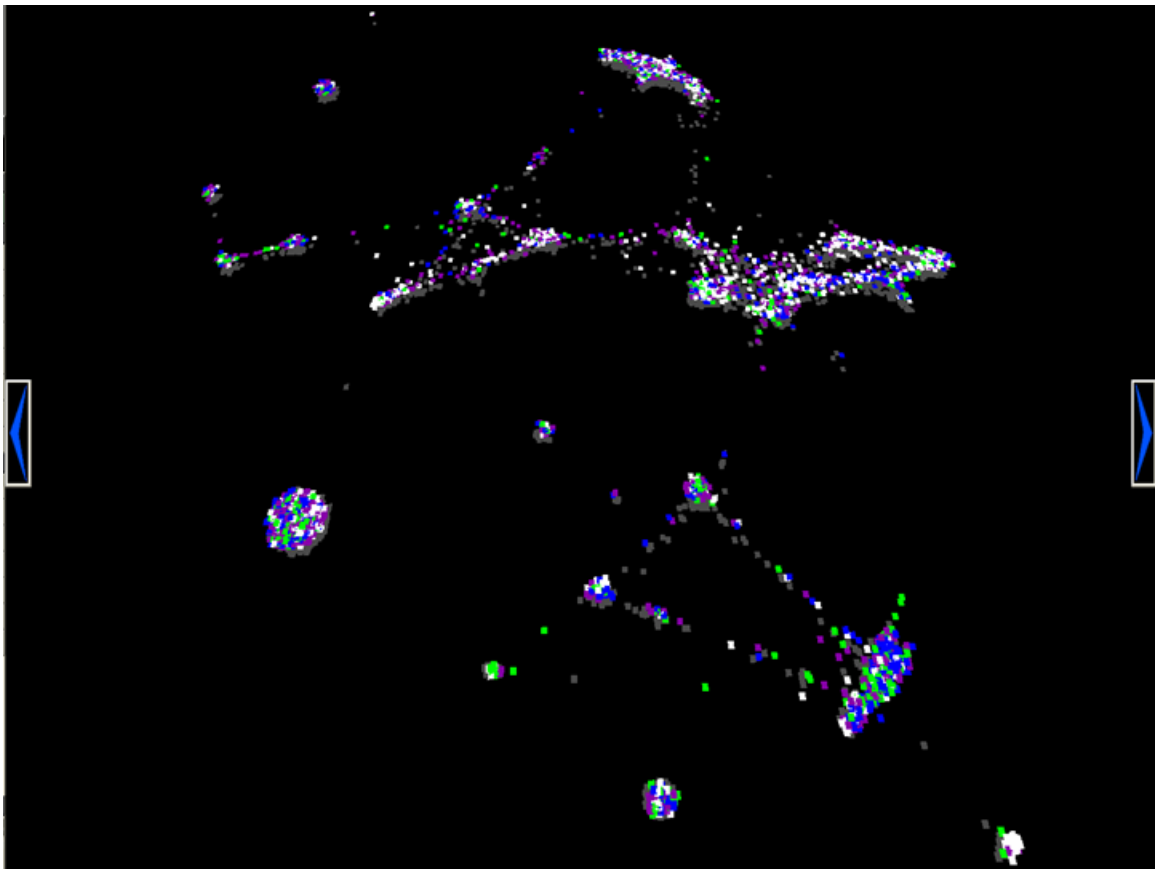


Figure 21. In this figure the genes have been clustered by expression levels in 254 leukemia patients, and colored as in the previous Figure. Little similarity between the two figures can be discerned, even with finer scale investigations than these two.

Similarity algorithms

While Pearson's correlation has a breakdown point of 1/N (a single outlier can distort the statistic from one extreme to the other[30]), it is easy to compute and has been well accepted in the microarray community. Because savage scored expression values are bounded, the influence of outliers is less important. As a result the correlation coefficient is usually the basis of our similarity computations. When too few arrays are available to have confidence in R, the percentage-bend coefficient[31]can be used instead.

It is common to cluster directly with these coefficients. However, doing so ignores much of the available information because R is such a nonlinear function. For example, there is a slight change in significance when comparing two pairs of genes that have R=0.5 and R=0.55, respectively, but the relative significance between R=0.90 and R=0.95 can be quite large. Consequently, it is better to transform these correlations to a measure of their relative

significance, which can be done by converting to the t-statistic for the observed correlation, R, between the pairs of values[29]:

$$t = \frac{R\sqrt{n-2}}{\sqrt{1-R^2}} \ .$$

Both R and t have computational issues that should be addressed. In particular, R is undefined when the variance of either X or Y vanishes, hence a minimum, acceptable variance must be determined. We typically require that

$$\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(y_i-\bar{y})^2 > 0.0001 \,,$$

otherwise, no correlation is computed for the pair of expression vectors. A related issue occurs with t when R approaches ± 1.0 too closely; the t-statistic becomes arbitrarily large. Because clustering will be comparing similarities, the strength of an extreme outlier will distort the clustering. Hence t should be clipped to avoid such extremes.[2]

Missing data continue to present concerns in computing R. Certainly, if too many values are missing, any computed similarity would be suspect. Recourse to the analyst's experience and judgment is the best way to choose how many values can be missing before the comparison is not attempted.[3] Computing all of these correlations produces a huge file of similarity comparisons. For example, the computation for an experiment[33] around *C. elegans*, which has about 20,000 genes, required the computation of about $2x10^8$ correlations. Using all of the correlations for clustering is neither necessary nor desirable. Most of the correlations will be modest and including them slows the clustering computation and introduces a great deal of resistance to the process that separates the mass of genes into clusters. For example, if some particular gene has strong correlations with a few tens of other genes, they should eventually be clustered together. However, if there are hundreds or thousands of correlations weakly linking that particular gene to others, the net sum of these weak correlations may overwhelm the few strong correlations.[4]

If only a few of the correlations will be used for clustering, some method of choice is required. The analyst can use all correlations above some threshold, or just the strongest few correlations for each gene. We have found the latter approach to be sufficient.[5] Interestingly, the distribution of the pairwise correlations offers some insight into the effect of the distribution rescaling discussed earlier.

---

[2] We typically truncate values greater than 300, though even this value may be extreme.

[3] For large collections of arrays, requiring that at least 70 measurements be simultaneously non-missing for both expression vectors has been acceptable in our experience.

[4] It can be argued that these weak correlations will sum to zero so that, in expectation, the effect of the strong correlations will still control the clustering. However, the variance of that sum will grow with the number of weak correlations used, thereby potentially masking the stronger similarities for any particular case.

[5] We have found that using the twenty strongest correlations is often a good starting point. However, even fewer correlations may suffice, especially with the methods discussed below for finding the most central ordination from a distribution of stochastic clustering results.

Figures, 22, 23, and 24 show the distribution of correlations between 254 Affymetrix arrays from a Leukemia study at the University of New Mexico (publications in preparation, 11/2003). Here the expression correlation is between arrays, not between genes across arrays.

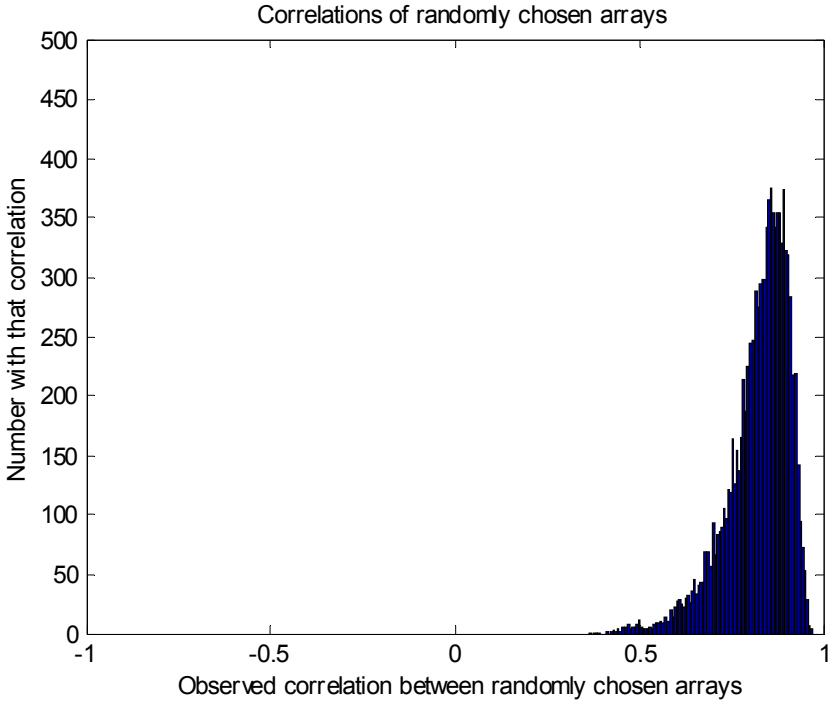


Figure 22. A histogram of correlations between pairs of arrays from 254 leukemia patients. These correlations are before any within-array normalization, and appear quite strong.

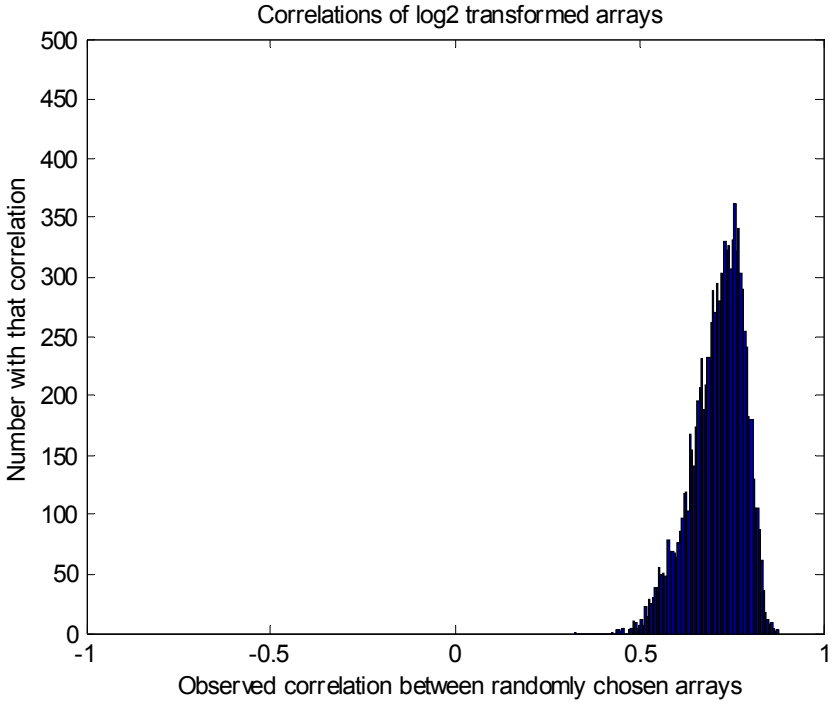


Figure 23. The histogram of correlations after transforming raw counts by taking logarithms. Note the mean correlation has apparently decreased.

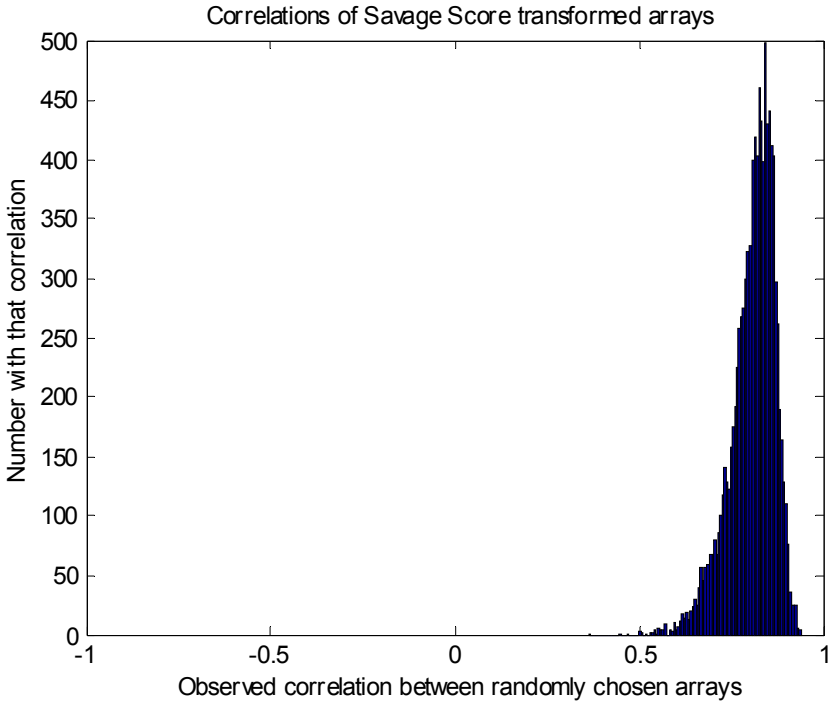


Figure 24. The array to array correlations after savage scoring the raw counts within each array. Note the more symmetric distribution and the further decrease in variance.

The arrays appear to be quite well correlated when the raw counts are used (see Figure. 22), but this correlation is illusory, the extremely high counts for a few genes are unduly influencing the correlations. If fact, *there should be some apparent differences between two groups of arrays in this study*, which contrasted patients with different treatment outcomes. If the arrays were, in fact perfectly correlated, it would be difficult to find differences in the gene expressions for these kinds of cancers, even though such differences are known, from other methods, to exist.

The arrays appear to be less correlated after the counts have been log transformed (see Figure 23). However, a heavy tail is apparent for the less correlated arrays. Finally, when the raw values are savage scored (see Figure 24) within each array, the array to array correlations are more symmetric. There is a slight increase in average correlation with respect to log transformed data, but the average correlation is less than is observed with the raw counts. Also note that the variance of these correlations decreases when savage scores are used.

Figures 25, 26, and 27 show cross plots of the correlations when using raw counts, $\log_2$ transformed counts, and savage scoring within each array. Correlations between transformed arrays (both log and savage scored) do not correlate particularly well with correlations computed using raw counts. This is believed to be the result of the strong influence of the few large raw measurements. On the other hand, there is good correlation between log transformation and

savage scoring, with a slope of nearly 1.0. However, the correlations based on logs are slightly less correlated than with savage scoring.

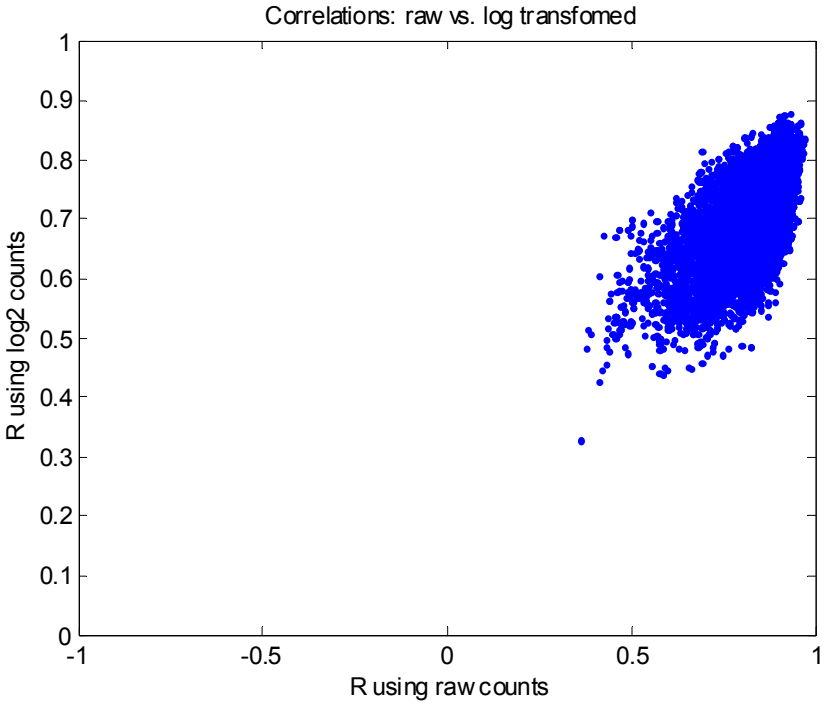


Figure 25. A cross plot examining the effect of log transforming the raw counts.

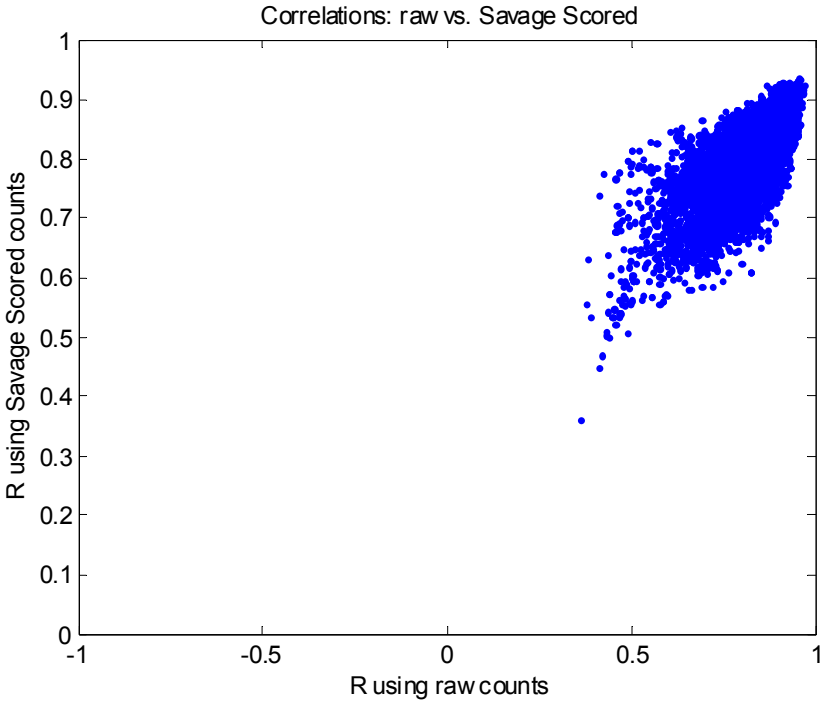


Figure 26. A cross plot examining the effect of savage scoring on the correlations.

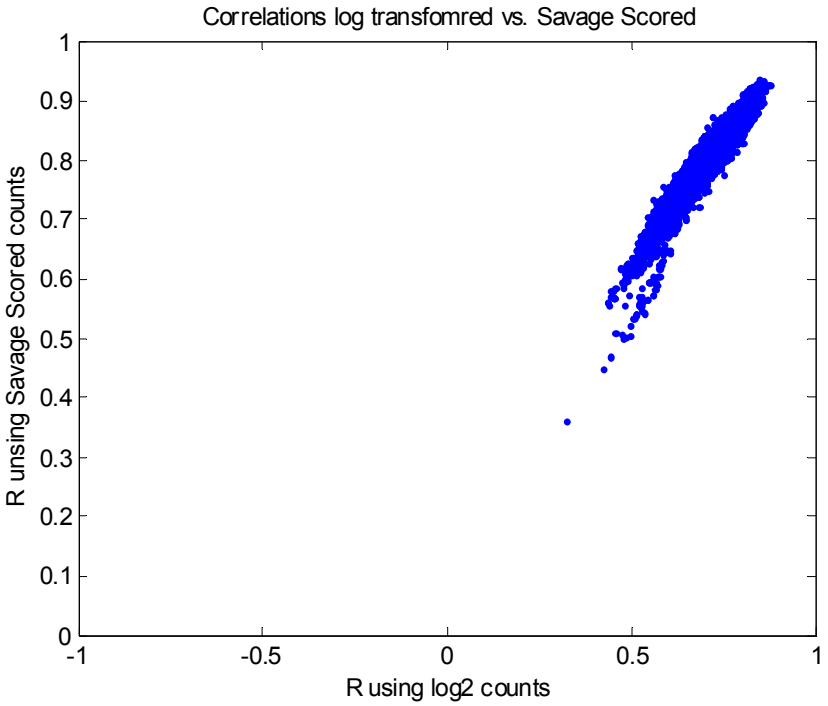


Figure 27. A cross plot comparing the effect of log transformation with that of savage scoring.

Figures 28, 29, and 30 show the distributions (with raw counts, log transformed counts, and savage scored counts) for these same gene correlation data; in this case *the correlations are between genes and across arrays*. These gene pairs were randomly drawn from the 8934 possibilities; hence there is little reason to anticipate any actual correlation between the selected genes (though, of course, it is not impossible that some pairs might be correlated). With this random selection in mind, on would expect a distribution centered about zero, and the smaller the variance about zero the more certain one could be about the actual correlations being zero, as expected. In fact, savage scored arrays do have a slightly smaller variance around zero, further supporting the use of savage scoring over log transforms. Interestingly, however, little difference can be seen between the use of raw counts and the use of log transformed counts, perhaps because it is unlikely to randomly select those few genes with extremely large raw counts.

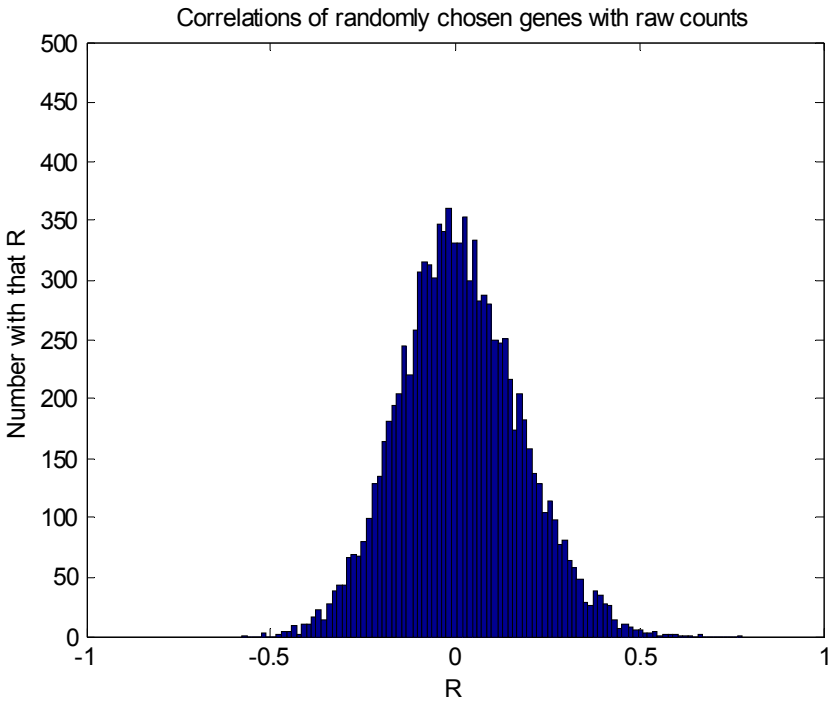


Figure 28. Correlations between randomly chosen pairs of genes using raw counts.

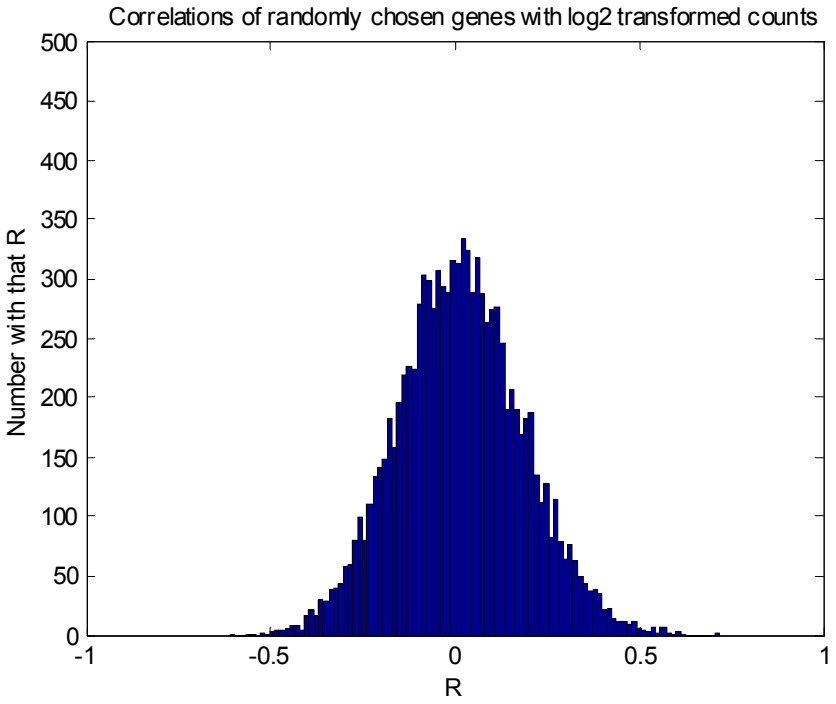


Figure 29. Correlations between randomly chosen genes when log transforms are used.

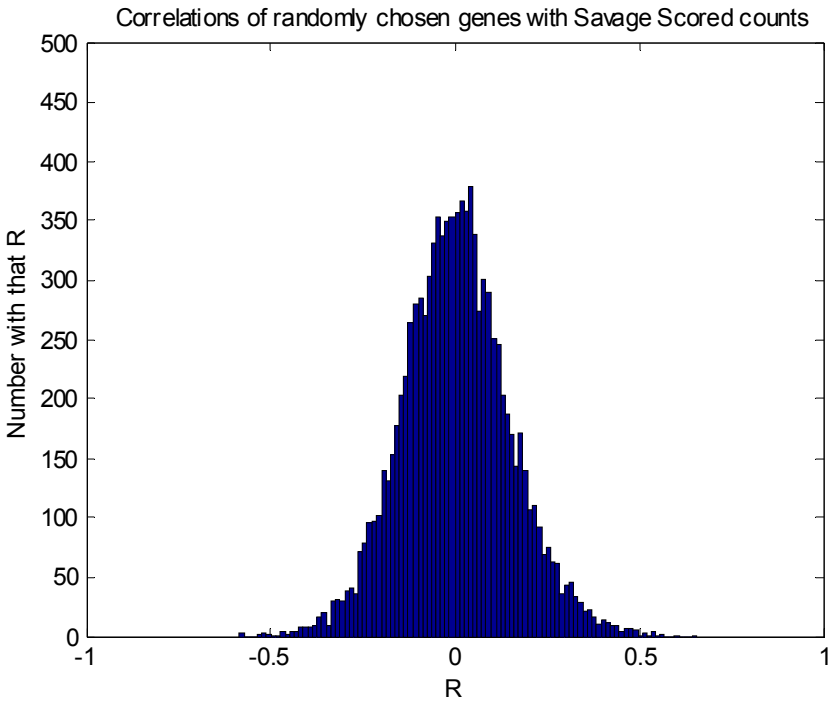


Figure 30. Histogram of correlations between pairs of randomly chosen genes when savage scoring is used. Note the slightly smaller variance when savage scoring is used, compared to either raw or log transformed data.

Figures 31, 32, and 33 show the cross plots for these gene correlations, which show little effect on R with respect to the use of either raw counts and log transformed counts, while the use of savage scores does narrow the range of observed correlation values.

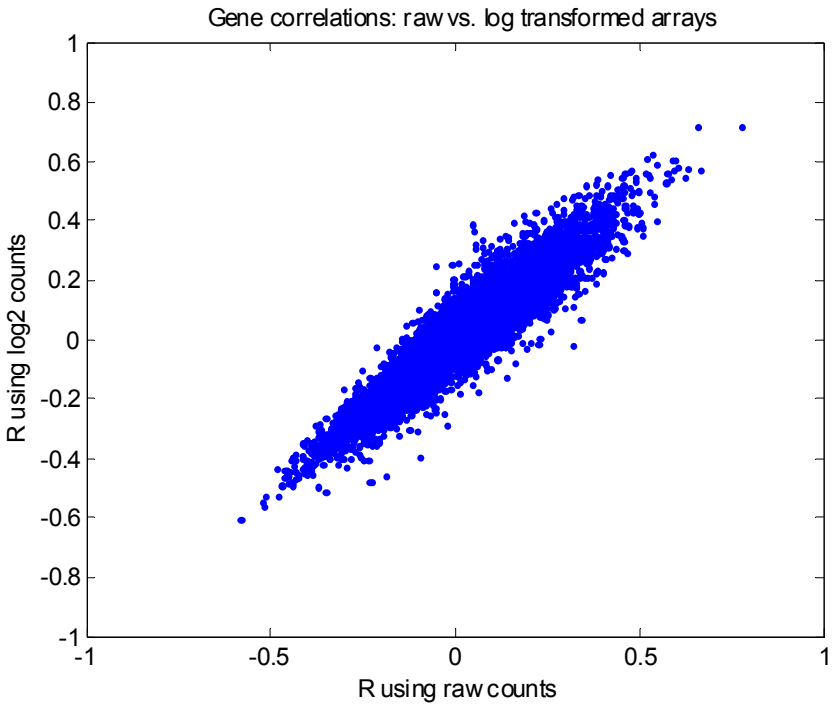


Figure 31. The cross plot suggest little the effect between using raw counts vs. log transformed counts for correlations between randomly drawn pairs of genes. On average, no correlation is expected.

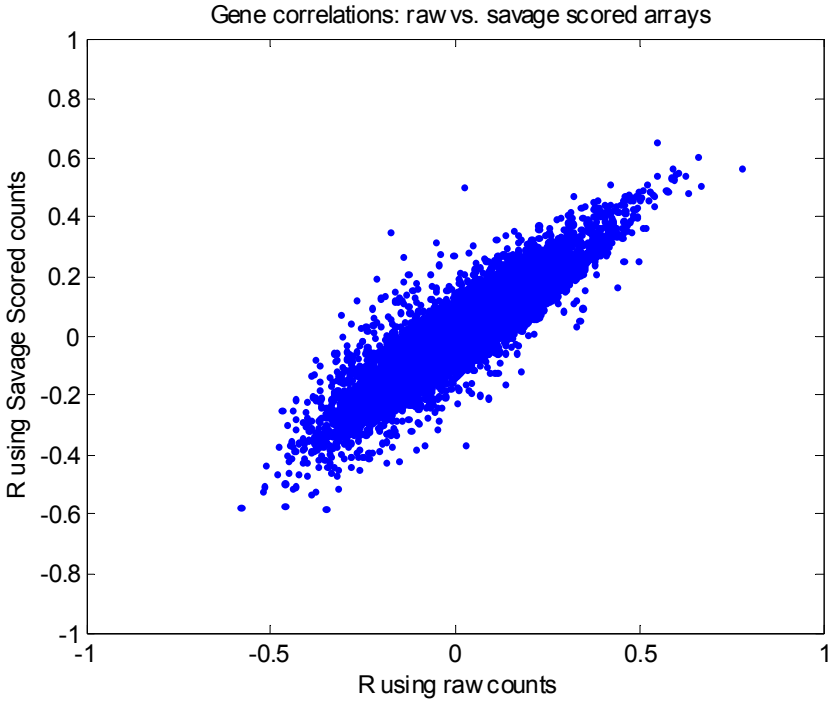


Figure 32. Cross plot comparing the effect correlations between randomly selected genes when using raw counts and when using savage scored counts within arrays.

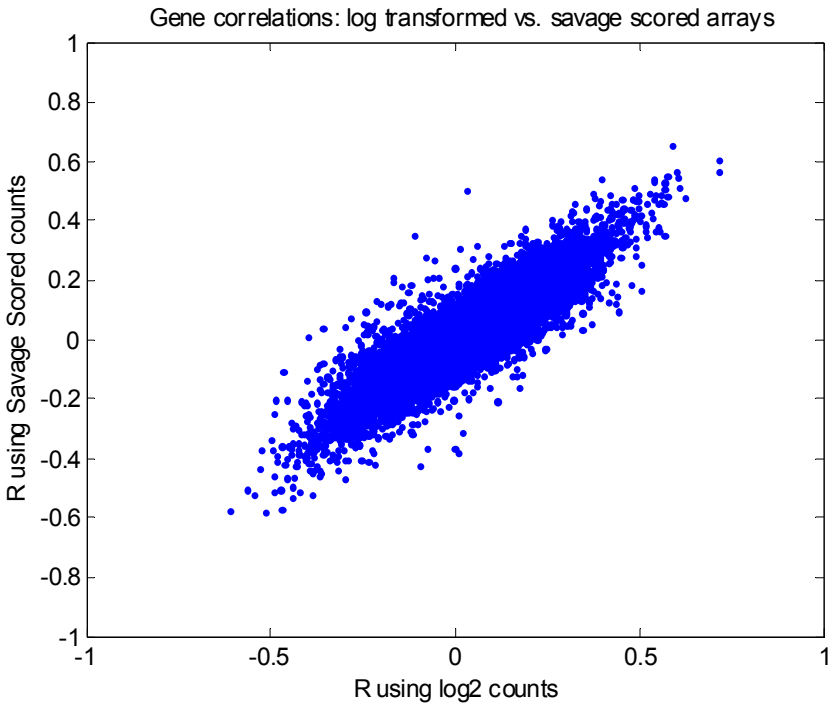


Figure 33. A cross plot showing the effect on correlations when using log transformed counts vs. the savage scored counts.

Processing measurements to create similarity connections

For a typical microarray experiment we cluster the gene using savage scored order statistics within each array after removing genes with too many missing values. Then pairs of genes are compared using Pearson's R. Generally, we will use only the strongest twenty positive correlations for each gene. These similarities are transformed as discussed earlier to the corresponding t-statistic to correctly reflect the significance of the correlations between the pairs. That is, we emit a similarity file having twenty entries for each gene, and each of these entries will have the name of the two genes and the t-statistic from the associated Pearson's correlation. This file serves two purposes. First, it is the input for the clustering algorithm, which will be described in the next section. Second, pairwise similarities can be visually examined, using the VxInsight connection list feature, to evaluate the quality of the clustering.

Connection lists

When a connection list is displayed in VxInsight on top of a visual cluster of the genes, pairs of related genes are shown with a line connecting them. This visualization is a quick way to evaluate how well the clustering algorithm performed. In particular, one would like to see a great density of connections between the genes in local clusters and fewer connections between genes in different clusters. Because we constrain the clusters to exist in two dimensions, it is generally

not possible to perfectly meet this criterion, and one should expect considerable cluster-to-cluster interconnection. Importantly, these inter-cluster connections reflect the larger scale structure of the clustering, as these are the similarities which tie various clusters together into related groups of subclusters. The strength of these groupings can be immediately visualized by the density of the similarity connections, see the VxInsight discussion.

Strongly similar connection lists

While the entire similarity file is used for clustering and can be visualized to evaluate the resulting cluster structures, one often would like to see only the *strongest* similarities to make sure that they are particularly localized within individual clusters. However, this concept of *strong similarity* is not well defined, and a range of acceptable criteria should be explored. The approach we use is, again, based on the analyst's experience and intentions.

The analyst specifies a particular correlation that is to be considered "strong," say a true population correlation of $\rho > 0.9$. Certainly, all observed sample correlations, R, above this value will be written to the strong similarity file. However, by random chance a pair of genes with actual correlation $\rho > 0.9$ may have experimental measurement with a sample correlation that falls below this threshold. The analyst controls this risk not by lowering the definition of the acceptable true correlation required to specify *strong correlation*, but by specifying the risk that would be acceptable for missing a pair of strongly correlated genes given the random nature of making measurements. For example, the analyst may require $\rho > 0.9$, but would be willing to miss a pair at that level one time in twenty. Hence, because the sample correlation, R, can fall around ρ, the threshold for R will actually be less than the selected value, 0.9; how much less is specified by the acceptable chance of falsely rejecting a strong correlation (here the one chance in twenty).

Computing the threshold requires an estimate for the distribution of the sample correlation around the true correlation, ρ. The transformation from R to t discussed earlier is only valid when $\rho = 0$, and hence is not suitable here. However, R. A. Fisher[34, 35] has shown that $Z_R = 0.5 \times [\ln(1+R) - \ln(1-R)]$ is distributed approximately normally with mean Z_ρ and variance $\sigma_Z^2 = 1/(n-3)$. Hence the following transformation may be used to compute the critical threshold matching the analyst's specifications:

$$Z = (Z_R - Z_\rho)/\sigma_Z .$$

One must first find the normal deviate Z_β, which matches the specified risk (one in twenty times, or 0.05, in this case); $P(Z < Z_\beta) = 0.05$, or $Z_\beta = -1.64$.

Z_ρ, and σ_Z^2 must be computed before solving for Z_R:

$$Z_\rho = 0.5 \times [\ln(1+0.9) - \ln(1-0.9)] = 1.4722, \text{ and}$$

$$\sigma_Z^2 = 1/(n-3)$$

$$Z_{Rcritical} = Z_\rho + Z_\beta \times \sqrt{n-3}) .$$

Then, finally the critical value, $Z_{Rcritical}$, can be found by inverting Fisher's transform, to give

$$R_{critical} = \frac{e^{2Z_{Rcritical}} - 1}{e^{2Z_{Rcritical}} + 1}.$$

All of the pairwise similarities exceeding this critical value can be saved in a file, which will be used later when displaying the strong similarity connections, as desired. The next step, following these preliminary calculations, is to use the similarities to cluster the genes. It should be noted, that this discussion has focused on clustering the genes using gene expressions across n microarrays. However, the same kinds of similarity calculations can be used when clustering together the microarrays, or the patients associated with those arrays. We generally compute both clusters, as they yield complementary information, and both offer visual clues about the clustering results. For example, Figure 34 shows a few of the 120,000 similarity links used to cluster the data; one can see that some similarities must be stretched in the course of laying out the entire graph. Figure 35 shows a close up of one of those clusters to show the much greater density of similarities within the clusters than between them. This is particularly apparent when strong similarities are used, see Figures 36 and 37. This use of connection lists displayed over a cluster of data is a very powerful visual tool, and is not restricted to just similarity connections. For instance [36] used this feature to simultaneously visualize gene expression data through the cell cycle with connections between genes whose gene products are known to interact.

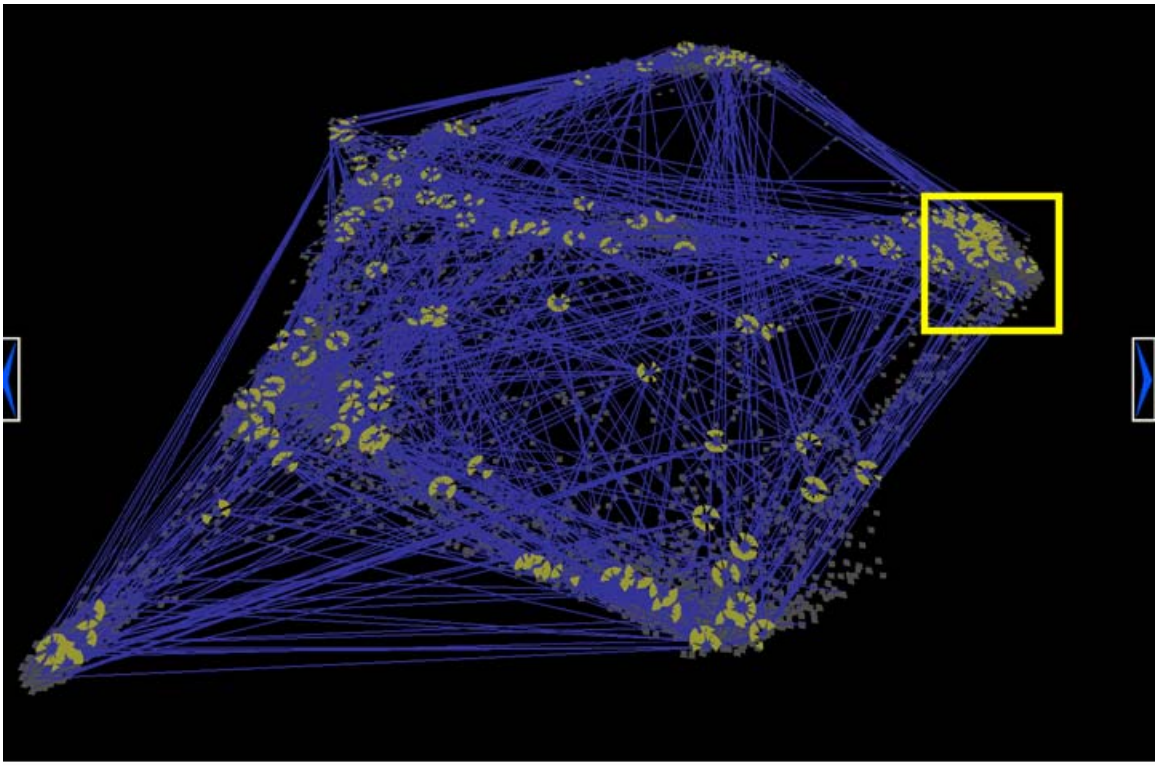


Figure 34. A few of the 120,000 links used to create the clusters. The marked region is shown in more detain in the next figure.

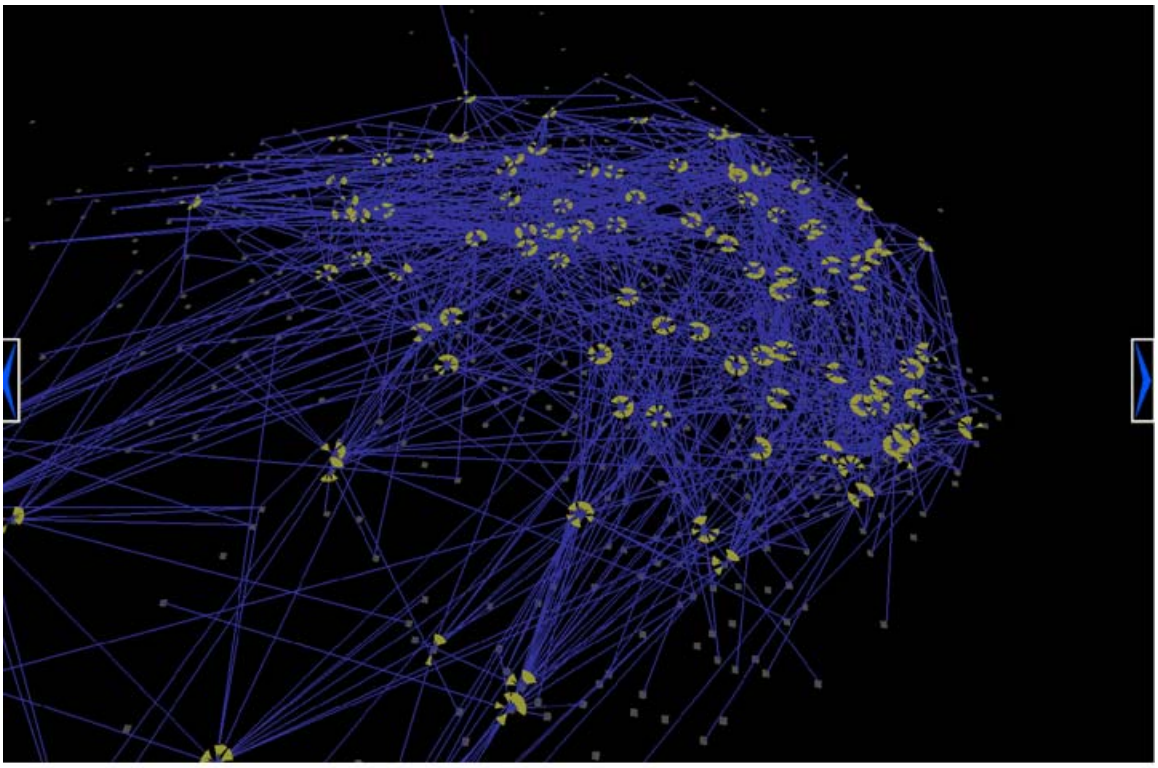


Figure 35. The links within the cluster indicated in the previous figure. Note the high density of links within clusters relative to between them.

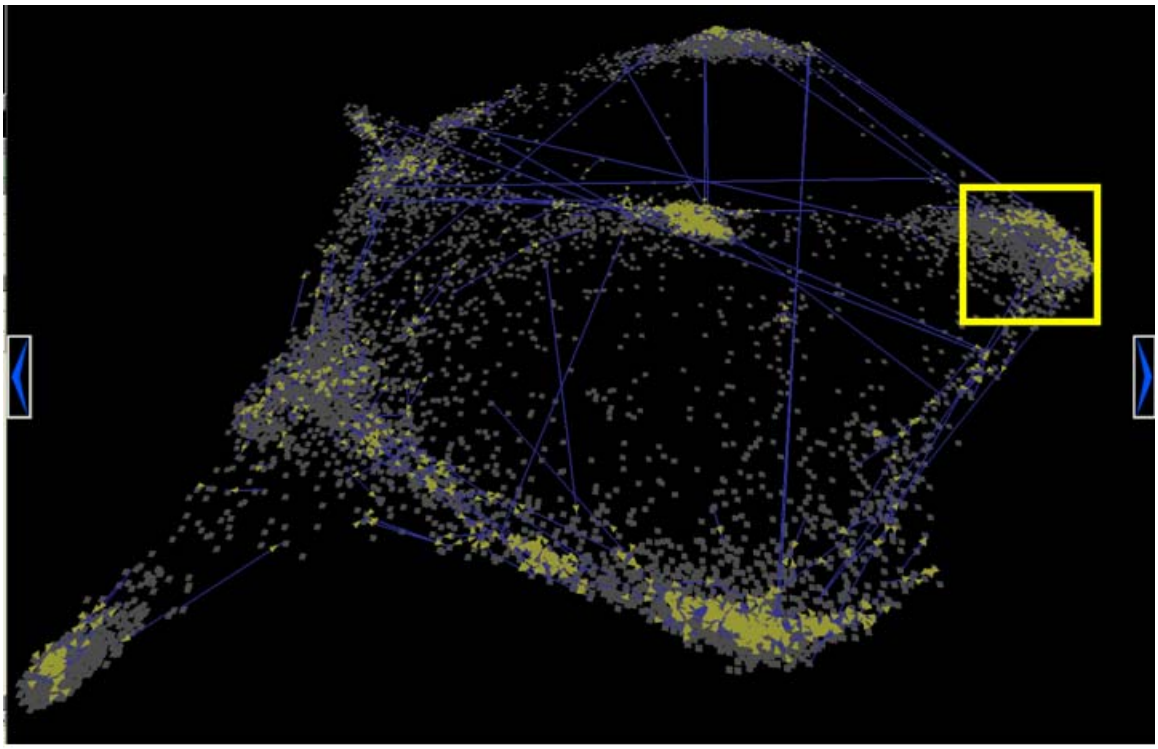


Figure 36. Just the strongest links across the whole cluster. The next figure shows a closeup of the indicated region.

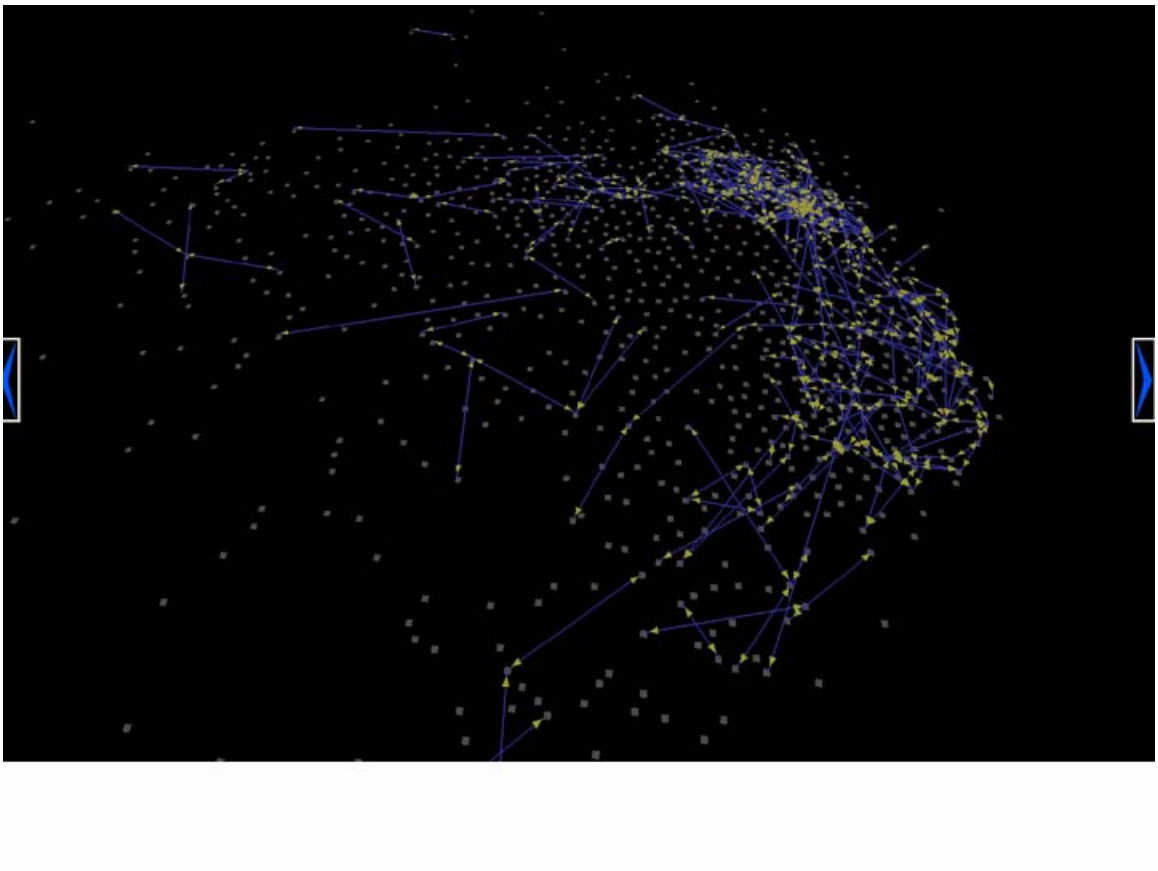


Figure 37. The strong links within the indicated region in the previous figure. Note the very high concentration of strong linkages within the cluster relative to those between clusters.

Clustering with VxOrd

The VxOrd clustering algorithm assigns two dimensional coordinates to vertices in a connected, weighted graph, where the edge weights are the non-negative similarities between the connected vertices.[6] The algorithm places genes into clusters such that the sum of two opposing forces is minimized. One of these forces is repulsive and pushes pairs of genes away from each other as a function of the density of genes in the local area. The other force pulls pairs of similar genes together based on their degree of similarity. During each iteration of the algorithm, small adjustments to the assigned coordinates are made in the direction minimizing the force on the gene being moved. The details of this implementation are critical, and have been described in detail in a previous paper[26].These details are reproduced below, placing the clustering algorithm within the larger processing and analysis context.

An abstract, edge-weighted graph, G = (V, E), is generated using a list of nodes and their similarities, where the vertices, V, correspond to the data objects, and the similarities correspond

[6] For convenience here, the vertices are assumed to correspond to genes, but they could equally well be arrays or patients.

to the weighted edges, E. An extensive literature exists for graph drawing and layout algorithms[37-45]. The work of Fruchterman and Reingold [38] is particularly relevant to our approach.

In developing and implementing our algorithm we were guided by four important principles:

1. Vertices connected by an edge should be drawn near each other.
2. Non-connected vertices should be forced away from each other.
3. The results should be insensitive to random starting conditions.
4. The complexity of computation should be reduced to a minimum.

These principles are so important that we will address each of them in detail.

Principles 1 and 2

Fruchterman *et al.* compute a 'force' term for both attraction and repulsion. These terms are then used to generate new positions for the graph vertices. Our algorithm combines the attraction and repulsion terms into one potential energy equation. The first term, in brackets, is due to the attraction between connected vertices; the second term is a repulsion term.

$$K_{i(x,y)} = \left[\sum_{j=1}^{n_i} \left(w_{i,j} \times l_{i,j}^2 \right) \right] + D_{x,y}$$

$K_{i(x,y)}$ = The energy of a vertex at a specific x, y location
n_i = The number of edges connected to vertex i
$w_{i,j}$ = The edge weight between vertex i and the vertex connected by edge j.
$l^2_{i,j}$ = The squared distance between vertex i and the vertex at the other end of edge j.
$D_{x,y}$ = A force term proportional to the density of vertices near x,y.

In our ordinations, the energy equation is gradually minimized in three phases in an iterative fashion. The first phase reduces the free energy in the system by expanding vertices toward the general area where they will ultimately belong. The next phase is similar to the 'quenching' step that occurs in simulated annealing algorithms, the nodes take smaller and smaller random jumps to minimize their energy equations. The last phase slowly allows detailed local corrections while avoiding any large, global adjustments.

All movements are random; each vertex is allowed to 'jump' from its current position to a new, random location. If the move reduces the potential energy for the vertex then the vertex is allowed to stay at the new location. Otherwise, the vertex remains where it was until the next iteration. Other, more complicated techniques, including gradient descent and methods with momentum terms, are theoretically appealing. However, the energy 'surface' for thousands of vertices is so chaotic (both spatially and temporally), that, in practice, we have found the simpler method performs better. Notice that for each vertex only its own energy is considered, a characteristic of a 'greedy' algorithm, which only indirectly leads to a global minimization for the entire system. However, the total energy of the system,

$$G = (V, E) : TotalEnergy(G) = \sum_{i=1}^{|V|} K_i ,$$

can still be used as a criterion for algorithm termination.

The literature [37, 42, 43]discusses many other termination criteria, some of which do not explicitly follow the total energy. Eades[37], for example, suggests simply running a fixed number of iterations, in their case 100. We have found that 800 iterations work well for our more complex graphs. We typically deal with graphs having on the order of 10,000 vertices.

Clearly, minimizing the potential energy should lead to ordinations that are consistent with our first two principles. The attraction term rewards movements that minimize the edge lengths between strongly weighted vertices. The second term, $D_{x,y,}$ which is a force based on the local density of nearby vertices, is minimized when vertices move to less crowded areas. In order to reduce both terms, a vertex must be close to its connected vertices and at a distance from non-connected vertices.

Principle 3

A stochastic ordination process can easily start in ways that prevent smooth transitions to correct answers. That is, the algorithm can get trapped in local minima, and is likely to be forced toward local minima early in the computation. The problem is that an initial configuration can result in some vertices that belong near each other being initially separated by a large barrier. Various stochastic techniques are used to avoid this problem. For instance, simulated annealing allows a probabilistic decision to take moves that will occasionally actually increase the energy associated with the node. This technique allows vertices to overcome the barriers associated with local minima in the effort to find lower energy states. Upon examination of our energy equation it becomes clear that 'barrier jumping' can be achieved by directly solving for the location that minimizes the energy for a single vertex, which can rapidly move a node through an energy barrier. We have successfully used this analytical approach for avoiding local minima early in our algorithm. Achieving a favorable configuration early in the process, independent of the starting configuration, is essential for efficient ordinations that are consistent with our third principle.

We achieve this result by moving vertices in the direction specified by energy equation most of the time. However, to jump over energy barriers a small fraction of the vertices ignore the repulsion term and minimize the attraction term analytically. This is accomplished by computing a weighted centroid over all connected vertices. The vertex then 'jumps' to that computed centroid, regardless of any possible energy increase, as shown in Figure 38.

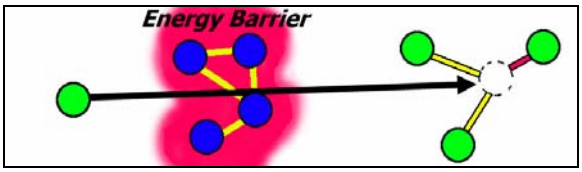


Figure 38. Barrier jumping by ignoring density term.

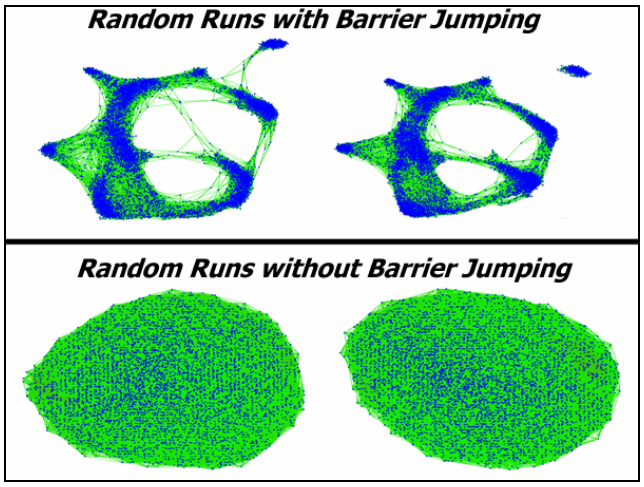


Figure 39. Two random runs with and without barrier jumping.

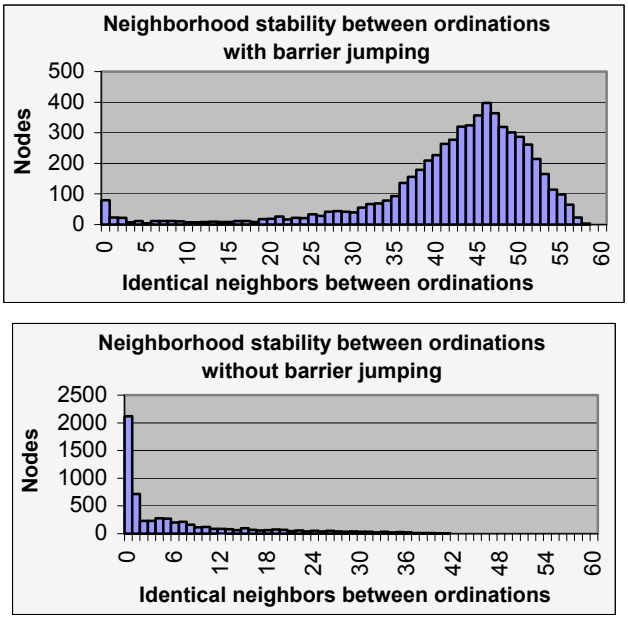


Figure 40. These histograms compare the random ordinations shown in the previous figure, with and without barrier jumping. Note that local neighborhoods are severely distorted without barrier jumping.

Barrier jumping is tied to the cooling schedule, and the frequency of barrier jumping linearly declines from 25% to 10% during the 'quenching' period and is not used at all during the simmer phase. The high frequency at the beginning is required for stability with respect to

random initial conditions. The poor initial placement or initial bad jumps that would otherwise irrevocably change the outcome of a purely random algorithm are greatly mitigated by the correcting nature of this process. Figure 39 shows images from two pair of random runs. Ordinations in the first row use barrier jumping, ordinations in the second row do not. We can see the excellent repeatability achieved by using the barrier jump technique. The second row shows that the 6000 vertices become hopelessly trapped in a web of local minima. The histograms in Figure 40 provide further support that barrier jumping improves the repeatability of the random iterative solver. The histograms measure the stability of the ordination algorithms by counting the number of identical 'neighbors' within the nearest 1% of the other genes The maps contain 6000 genes so for every gene, we measured how many of the 60 nearest genes remained the same between runs.

Principle 4

The brute force approach for computing $D_{x,y}$ is certainly not consistent with our fourth principle. Because each vertex would have to check its position against all other vertices; this unsophisticated approach would take |V| comparisons for each determination of $D_{x,y}$. As every node must compute $D_{x,y}$ when determining its energy at a specific location x,y, the algorithm would require total running time $\Theta(|V|^2)$.

For real world problems an $\Theta(|V|^2)$ algorithm is prohibitively expensive. We have developed a grid-based method for computing $D_{x,y}$ that allows each vertex to determine an *approximate* value for this term in constant time, $\Theta(1)$, thereby reducing the total running time to be a satisfactory $\Theta(|V|)$.

The grid-variant algorithm discussed by Fruchterman[8] uses a binning technique to consider only those vertices within a certain neighborhood. This is an approach that, with a uniform distribution of the vertices, will reduce the calculation to $\Theta(|V|)$. However, a graph will only have a uniform distribution if the number of edges is small. Highly connected graphs will have dense concentrations of vertices in small areas, and the run time is no longer linear with the number of vertices. To be effective for all graphs, our repulsion term utilizes a 'non-specific' density measure. Vertices are not repulsed by other *specific* vertices, but are repulsed by a general overcrowding. This minor modification to the repulsion criteria allows a dramatic reduction in computational complexity.

This *density field* algorithm is implemented by having each node place an energy footprint onto a two dimensional (density field) array. The energy footprint may be any function in two-space. Our implementation uses a circle with radius *r* and a function that peaks at the center of the circle, while falling off quadratically with increasing distance from the center of the circle. The total density field is the sum of the contributions of each vertex in the region. Given the density field, a node can determine an approximate $D_{x,y}$ value using a constant time table lookup method. This method reduces the computation of the repulsion term from $\Theta(|V|^2)$ to $\Theta(|V|)$, and is consistent with our fourth principle, an important result for using our algorithms with real applications.

Clustering parameters

There are three important controls the analyst has over the VxOrd algorithm:

1) The number of similarities used for the clustering.
2) The degree to which the algorithm is free to ignore, or "cut" a few similarities between genes, which would otherwise be strongly pulled toward multiple clusters.
3) The amount of effort that the algorithm is allowed to spend in trying to sort out the local structure of the clusters, once the global clustering has been computed.

The first control concerns how many similarities are passed to the clustering algorithm. Every gene has some correlation with every other gene; however, most of these are not strong correlations and may only reflect random fluctuations. By using only the top few genes most similar to a particular gene as it is placed into a cluster we obtain two benefits: the algorithm runs much faster, and, as the number of similar genes is reduced, the average influence of the other, mostly uncorrelated genes diminishes. This change allows the formation of clusters even when the signals are quite weak. However, when too few genes are used in the process, the clusters break up into tiny random islands containing only two or three very similar genes, so selecting this parameter is an iterative process. One trades off confidence in the reliability of the cluster against refinement into sub-clusters that may suggest biologically important hypotheses. These clusters are only interpreted as suggestions, and require further laboratory and literature work before we assign them any biological importance. However, without accepting this trade off, it may be impossible to uncover any suggestive structure in the collected data.

As an example of the impact of these parameters, consider Figures 41 through 43. Here we are clustering a set of 126 arrays, each with about 12,000 genes. First consider the effect of using too many similarities. Figure 41 shows the result when 100 similarities per array are used. In this case, there is a similarity connection from every array to all but the least similar 20% of the other arrays. With so many weak connections no clustering is apparent. The same is true when 30 similarities are used. However, when only the top 15 strongest similarities are used, two main groups begin to be apparent.

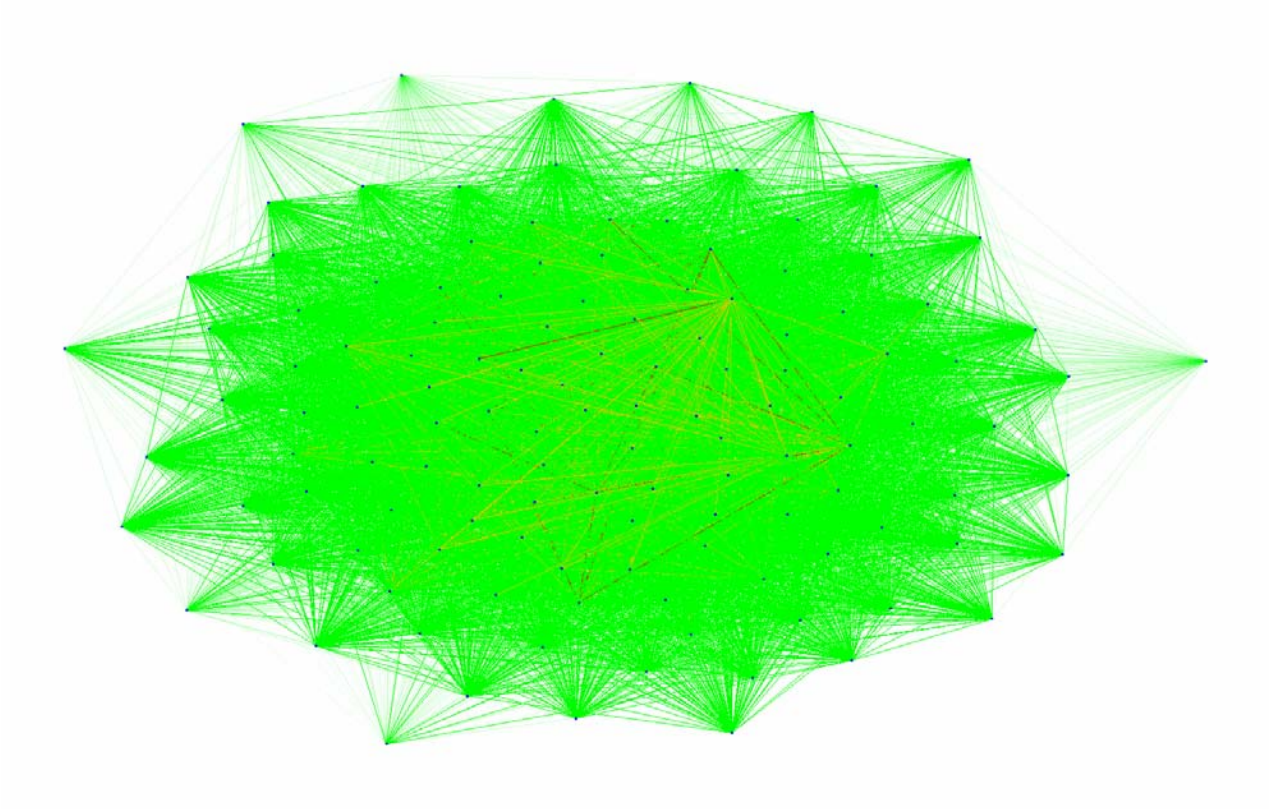


Figure 41. When using too many similarity links, 100 in this case, only a single undifferentiated group is formed.

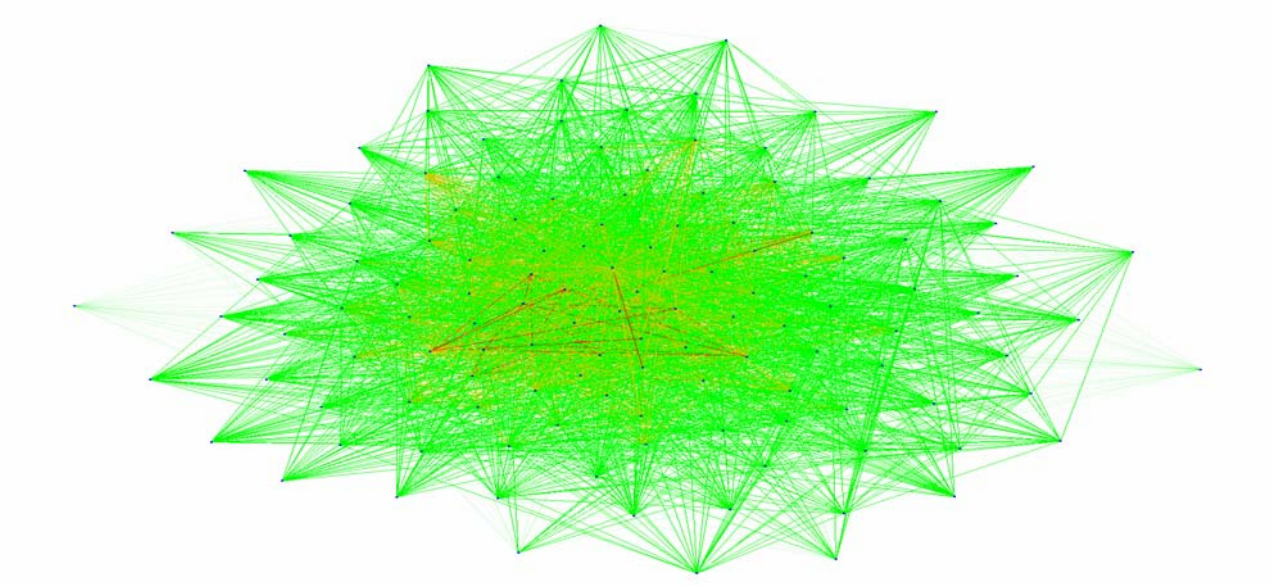


Figure 42. The same data clustered with 30 similarity links still does not separate into clusters.

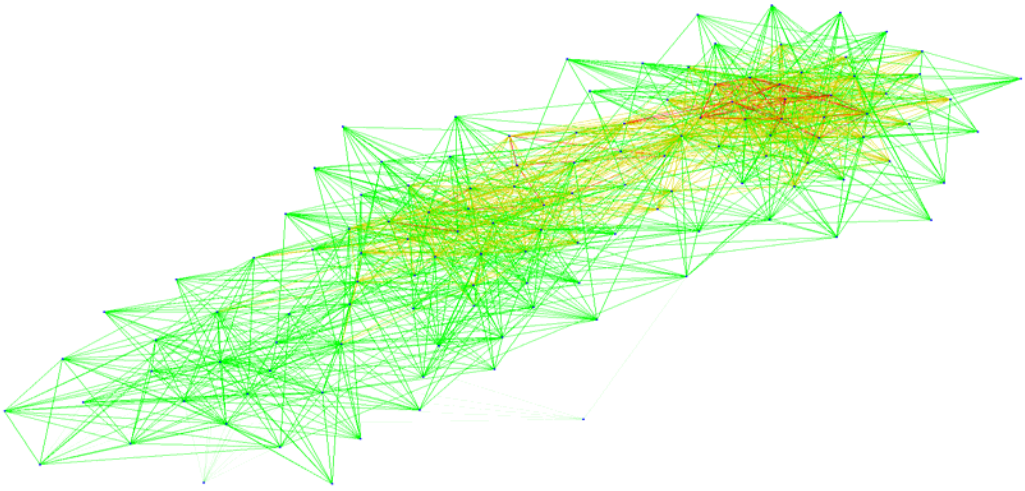


Figure 43. With only 15 similarity links the data is no longer completely undifferentiated, some stronger similarities are beginning to force the emergence of structure.

When a set of elements have a relatively uniform set of similarities it can be very difficult to separate them in to subclusters. However, there may be a few subsets of stronger similarities that could divide the data into clusters if these strong ones were allowed to express their influence in the absence of the other, mostly homogeneous, similarities. That is, small cliques of vertices may be revealed by removing, or cutting similarity relationships that have been constraining the vertices such that they remain in an undifferentiated agglomeration. Figure 41, and 42 show that no cliques are apparent when using 30 and 100 similarities per vertex for this group of 126 arrays, even with extremely aggressive edge cutting. On the other hand, the suggestive clusters seen in Figure 43 readily break into more detailed cliques when only 15 similarities per vertex are used, and when aggressive edge cutting is enabled.

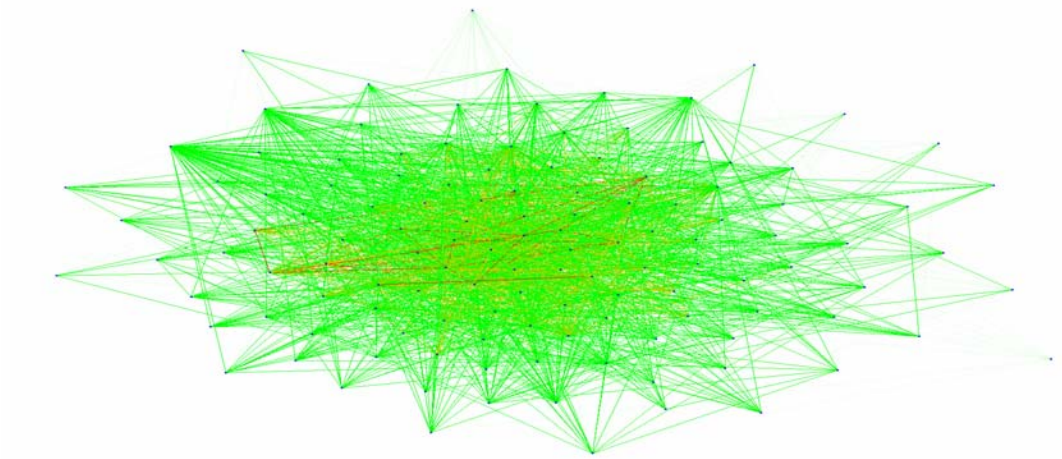


Figure 44. Here 100 similarity links were used to cluster and the most agressive edge cutting setting has been selected, which cannot overcome the effect of using too many links.

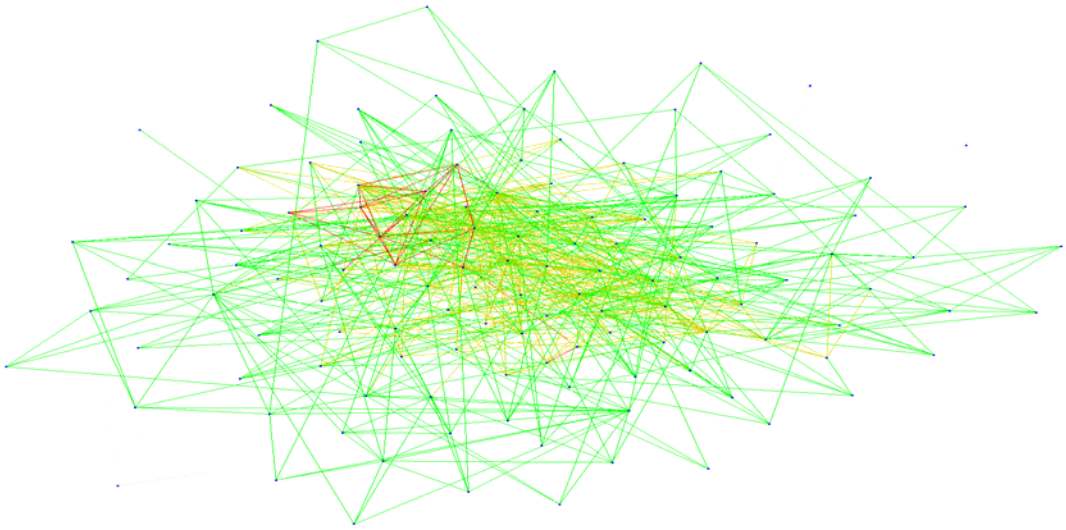


Figure 45. Here only 30 links were used and the maximum edge cutting has been enabled, but clusters are still not apparent.

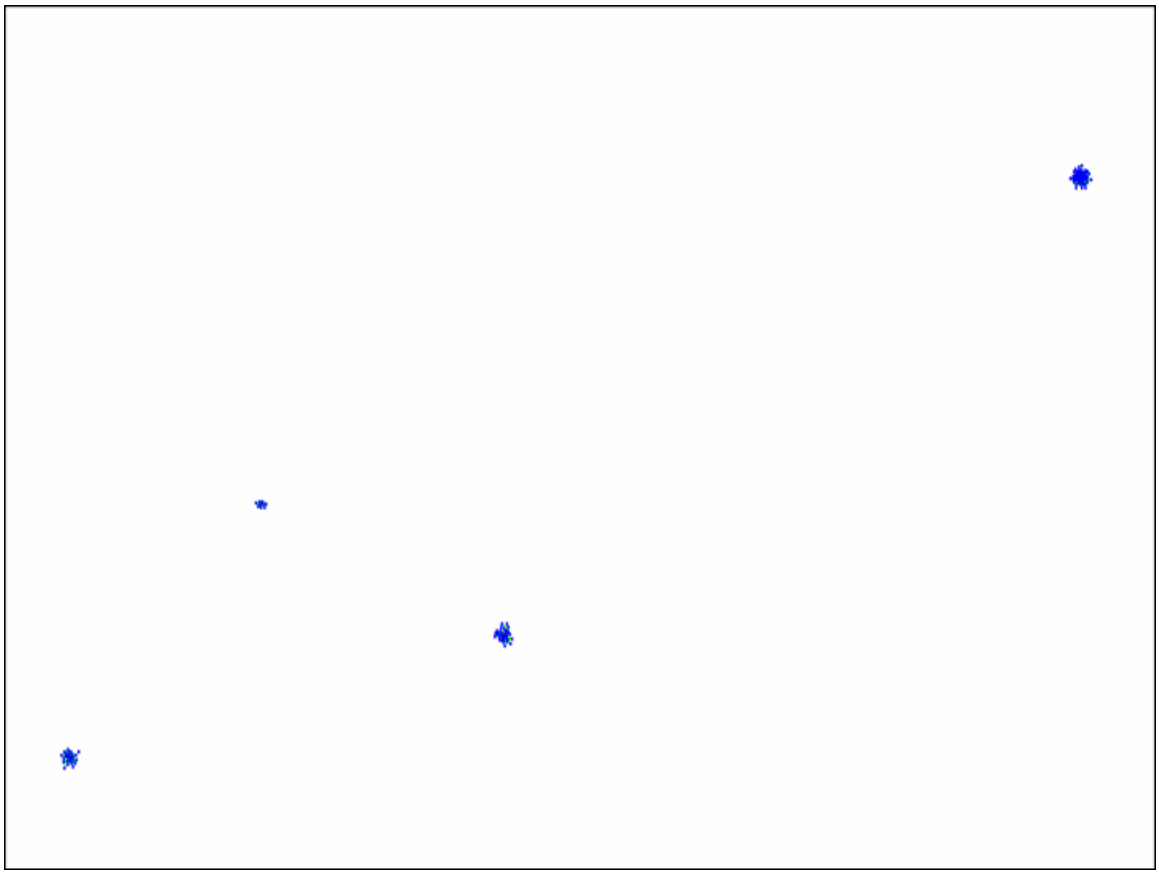


Figure 46. Finally, with 15 similarity links as in Figure 43, and after aggressive edge cutting, the data are able to separate.

At the expense of further processing, each of the cliques, or subclusters, revealed in Figure 46 will organize into structures with greater separation and internal order. For example, Figure 47 shows the results of allowing this subclustering. Interestingly, one of the clusters remains much more tightly gathered together than the others, which suggests the elements in this cluster are more similar to each other than the elements in other clusters are to each other.

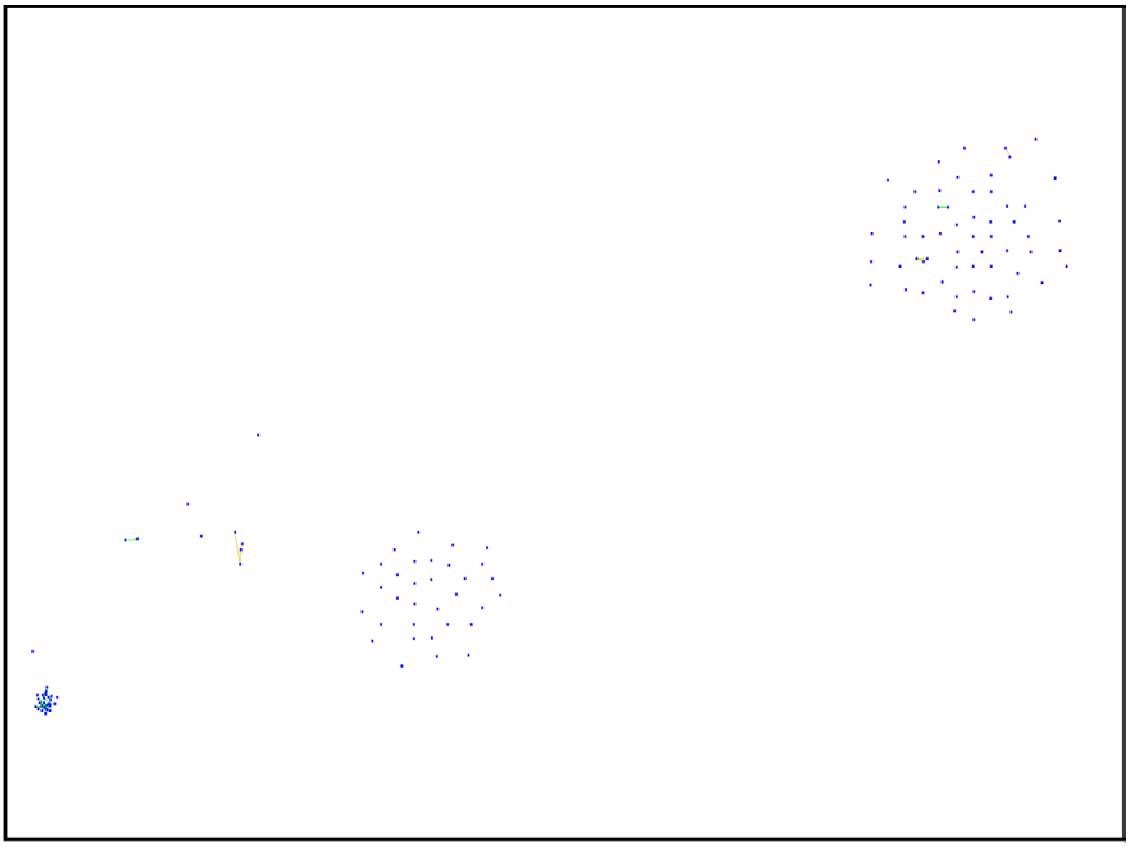


Figure 47. With 15 similarity links, maximum edge cutting, and extra subclustering iterations, the clusters are well separated and individually dispersed.

Evaluating the utility and significance of the clustering

Clustering algorithms are designed to find clusters. However, one's initial stance should be that there is no reason to suppose that the found clusters are more than artifacts. The very first evaluation should be an investigation of the clustering algorithm using exactly the same processing parameters, but with randomly permuted versions of the measurements. If the clustering algorithm finds clusters or structures in this randomized data then the results with the original data should be suspect. The processing methods discussed above have been tested in this way and randomized data do not exhibit any organized structure, see for example the analysis in [33], where the randomized data resulted in a single, symmetric, and otherwise unorganized group of genes, see Figure 48, which shows both the revealed structure in the data and the lack of structure in the randomized data. If randomized data shows no structure, then the structures in the actual data become more interesting and may possibly be useful. We next address various ways to further evaluate the clusters' meanings and strengths.

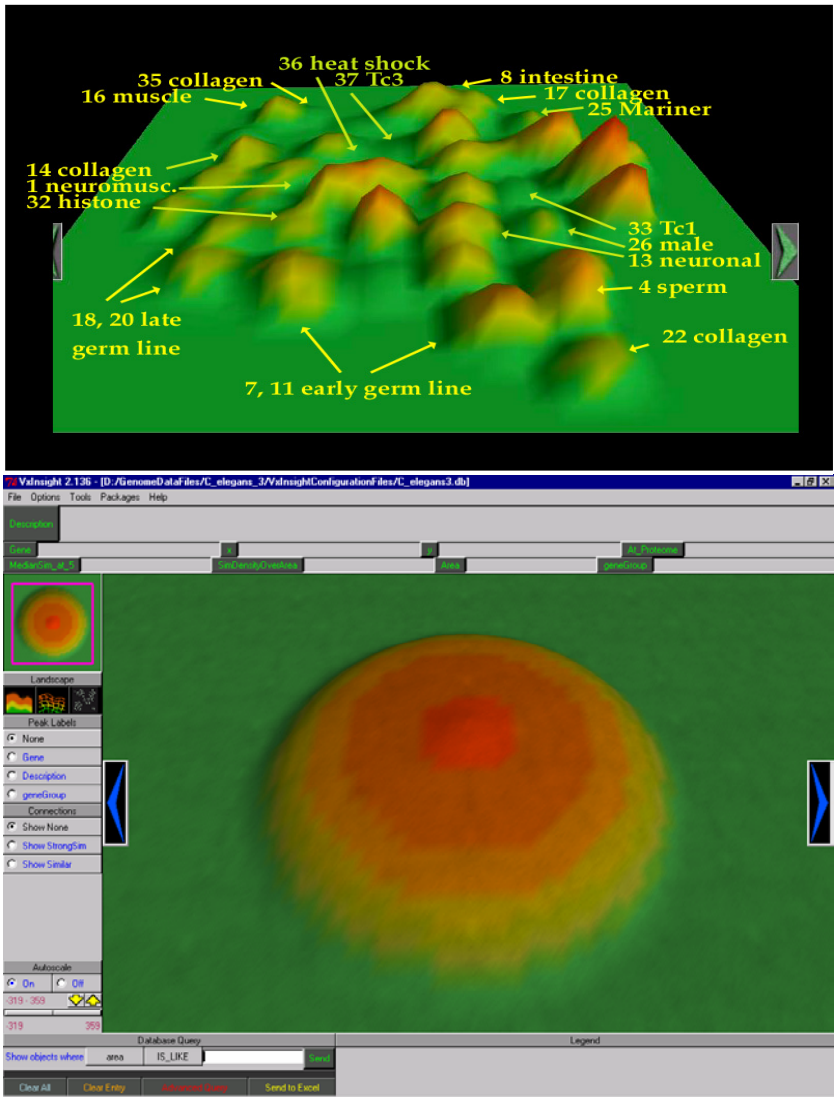


Figure 48. Clusters of *C.* elegans genes using 20 similarity links (top) followed by the result when the data values have been randomly shuffled (bottom), which show no differentiating structure.

Interactive exploration of the clusters

There are manual explorations that may increase ones confidence in the clusters, and there are more rigorous statistical techniques that should be applied. We often explore the clusters looking for collocated genes whose collocation makes sense biologically. For example, one of the earliest tests we performed was to assure ourselves that, for data from the Spellman cell cycle experiment[46], the closely related genes reported by Eisen[47] were collocated in our clusters, too. Figure 49 shows one particular set of genes that were strongly related in his study and are also very near each other in our clusters.

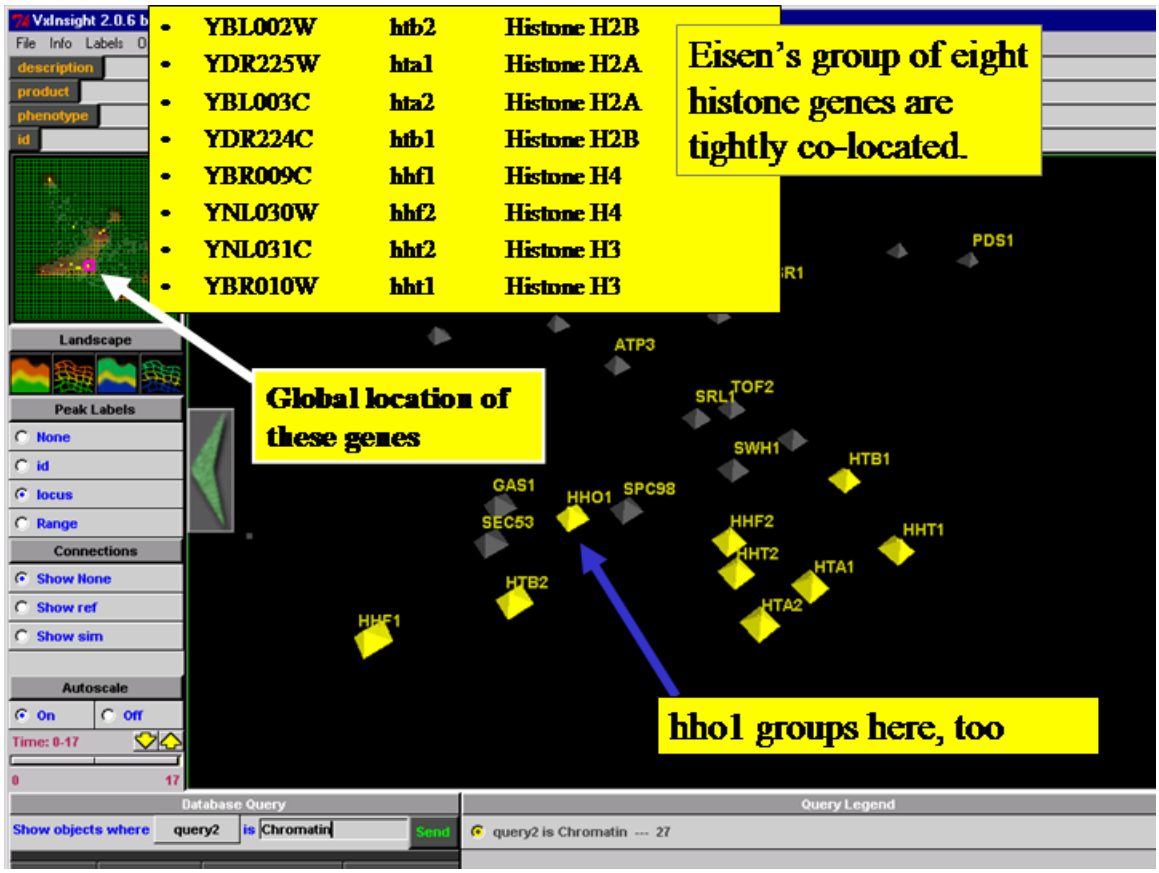


Figure 49. A typical quick check to verify that genes that are known to be related do in fact collocate.

Another sanity check compares the typical expression histories of the genes in each cluster to assure ourselves that genes in the cluster have, generally, uniform expression patterns, and that these patterns are different in the various clusters. While this is a visual inspection, the idea will be recast more rigorously in one of the statistical tests discussed later. Figure 50 shows Spellman's yeast cell-cycle data clustered with VxInsight overlaid with expression traces for typical genes in the various clusters. Not only do these traces seem homogeneous with the clusters, and different between clusters, but they also have biological significance as the cells move through their replication cycle. Surprisingly, the various states in the cell cycle correspond to a clockwise progression around the roughly circular layout of gene clusters in this Figure.

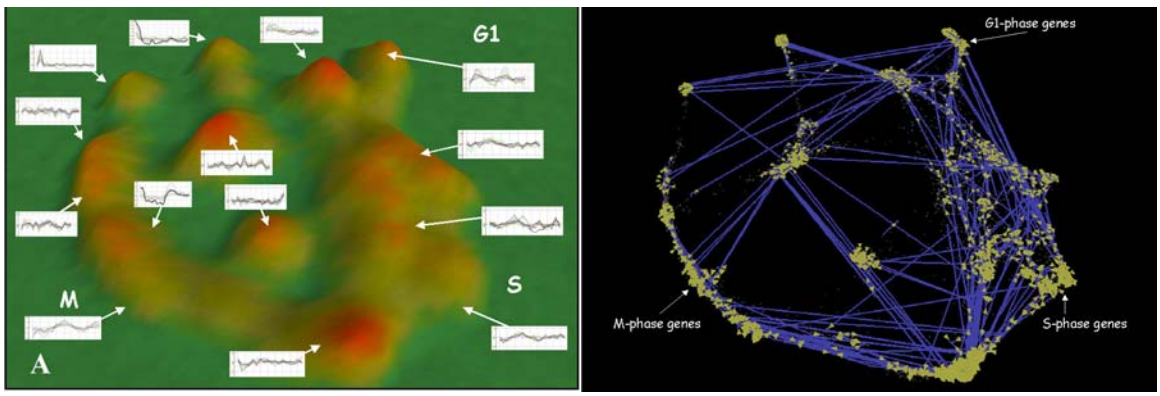


Figure 50. Cell cycle data with typical expression traces from each cluster (left). Interestingly, the clusters layout in a circle corresponding to the temporal cell cycle phases. Similarity links (right) are greatly concentrated within the clusters relative to between the clusters.

Statistical exploration of the clusters

These visual inspections are useful, but more rigorous statistical methods should also be applied. Here, two approaches are presented, one of which uses only the available expression data to compare correlation differences between and within clusters. The other approach makes use of externally available tables of genes which are known to be involved in the various biological processes. Genes involved in the same process are therefore assumed to be coordinately expressed, and should be, therefore jointly statistically enriched in some of the clusters.

Kim, *et al*.[33] tested the clusters of *C. elegans* genes for such enrichment and found significant statistical enrichments. These enrichments suggest that other genes in the same cluster could be expected to be involved in the indicated processes. This hypothesis was confirmed with laboratory experiments in several cases reported in that paper. Enrichment can be computed for a given list of n genes out of the total number of genes, N. If a cluster of M genes includes m of the n listed genes, then the relative enrichment is (m/M) / (n/N). An exact p-value can, thus, be computed assuming M independent Bernoulli trials each with probability, p=n/N.

We published in Werner-Washburn, *et al*.[36] another useful approach, motivated by the visual inspection of expression profiles in and between groups. There the question, "Are two mountains in the VxInsight map significantly different from each other?" is answered by comparing the empirical distribution of pairwise correlations in each mountain, and also the distributions of correlations between the two mountains. There are three ways clusters could systematically differ from each other:

- Expression correlations within each of the two mountains could be very different from each other, and also different from the inter-mountain correlations.
- The correlations might be vaguely similar in each of the mountains, but their inter-mountain correlations could be noticeably different from the correlations in either mountain.
- The correlations in each mountain could be noticeably different from each other, but the intermountain correlations could have some intermediate value, such that the intermountain correlations could not be detected as being different from either of the mountains, even if the mountains were, themselves, statistically different.

The first case corresponds to strongly separated clusters, the second to weakly separated clusters, and the third case corresponds to a gradual gradation from one cluster into another. However, there is only one way that the genes can be incorrectly separated into different groups: that is, if all three groupings are found to be indistinguishable.

If the gene expressions for genes in, and between, the two mountains were really indistinguishable (the null hypothesis), then analysis of variance (ANOVA) should fail to detect

a significant difference between the means of the three sets of correlations. We tested a number of clusters using ANOVA to assure ourselves that the clustering was significant.

Briefly, we started with two non-intersecting gene lists, GroupA and GroupB. We computed all possible correlations between the genes in GroupA, all possible correlations between genes in GroupB, and finally the correlations between every gene in GroupA with every gene in GroupB. These individual correlations were transformed to their corresponding T-statistics, which are directly related to the p-values associated with observing the correlations when the expressions are not actually correlated. Analysis of variance was performed to test if the mean correlations for these three different groups were significantly different. Under the null hypothesis, one would rarely see large F-statistics from this analysis. On the other hand, ANOVA should uncover a difference if the genes in the two VxInsight clusters were correctly separated into different groups. That is, we expect ANOVA to yield a very small p-value when the expressions for genes in either mountain are more like the expressions for genes in the same mountain than they are for genes in the other mountain. Further, when the correlations *between* the two clusters are different from the correlations in at least one of the mountains, ANOVA should also allow us to reject the null hypothesis. In either case we would conclude that the VxInsight clusters are not artifacts. This test was used in Werner-Washburne, *et al*. to show that a subset of genes associated with cell cycle phase G1 were collocated with $p<0.001$, and further that CLB6, RNR1, CLN2, TOS4, and SVS1 collocate with $p<0.0001$ for cells exiting from long-term stationary phase.

These reported methods have been useful in showing that the clusterings are not chance occurrences, and have led to scientific insights. However, these approaches have not addressed two other important issues related to clustering. First, how stable are these clusters with respect to variations in the measurements. Second, how stable are they with respect to different random initializations of the VxOrd clustering algorithm, which has an inherently stochastic nature. We turn our attention to these issues in the next section.

Stability of clusters

Much of the following work has been reported in[26], from which liberal quotes are extracted here. However, the subsequent analysis of the clustering algorithm to determine the most central ordination from a sample distribution of ordinations is new.

To test the stability of the algorithm to random starting points, we ran 100 re-ordinations of the Spellman cell cycle data[46], which had about 6000 genes. Each re-ordination was started with a different seed for the random number generator. We then visually marked the elements of a cluster in one ordination and looked to see if they were still visually clustered together in the other ordinations. We then computed the neighborhood statistics as described below.

To determine if small changes or noise in the similarities would give small changes in the ordination results we ran eighty re-ordinations where we added noise drawn from a gaussian distribution with mean zero, and standard deviations 0.001, 0.010, 0.050, and 0.100, and recomputed the ordinations (these noisy correlations were clipped to remain in the valid range of the correlation coefficient [-1.0, +1.0]. These different ordinations were compared, visually and statistically.

Evaluation methods

We compared the various ordinations using a neighborhood analysis. When two ordinations are very similar it is reasonable to expect that for every gene, the set of its nearest, say 60, genes would be almost identical in both ordinations. In fact, we would expect the same thing for every gene in the entire ordination. On the other hand, if the ordinations have almost nothing in common, it should be rare to observe a gene that had the same neighbors in both ordinations. We computed these neighborhood statistics for each gene, in each of the two ordinations. For each gene, we first identified the 60 nearest genes, and then counted of the number of genes in both neighborhoods. This number was used to increment the value in a table, so that in the end, we had a histogram showing how many genes had no common neighbors in the two ordinations, how many had one common neighbor, etc., up to the number of genes with exactly the same 60 neighbors in both ordinations, and histograms were prepared, as shown in Figure 51.

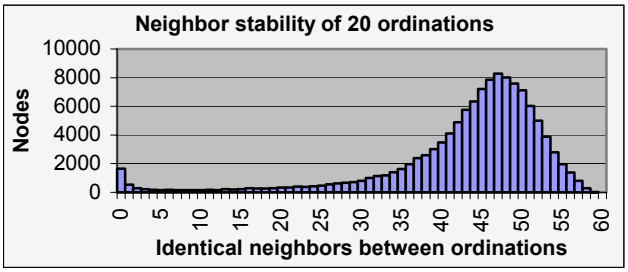


Figure 51. Distribution of neighbors between ordinations with random starting conditions, 20 replicates.

We then visually compared the results of the two ordinations by coloring all of the genes in a cluster found in the first ordination and seeing where those colored genes were placed in the other ordination (so that a similar ordination would not break up the group of colored genes, but would still have them co-located; see Figures 52 and 53).

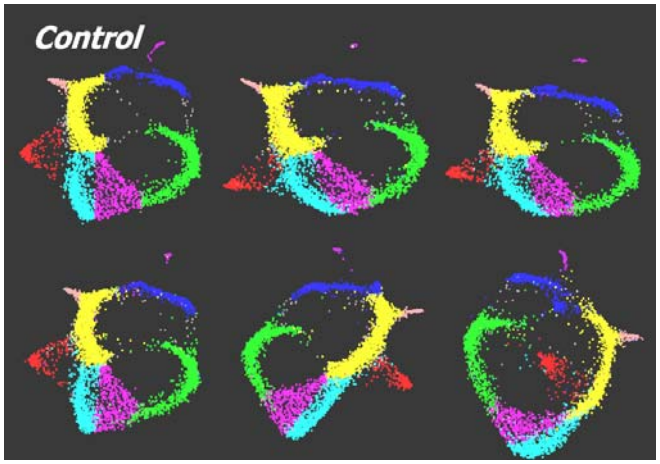


Figure 52. Ordinations with different random starting conditions.

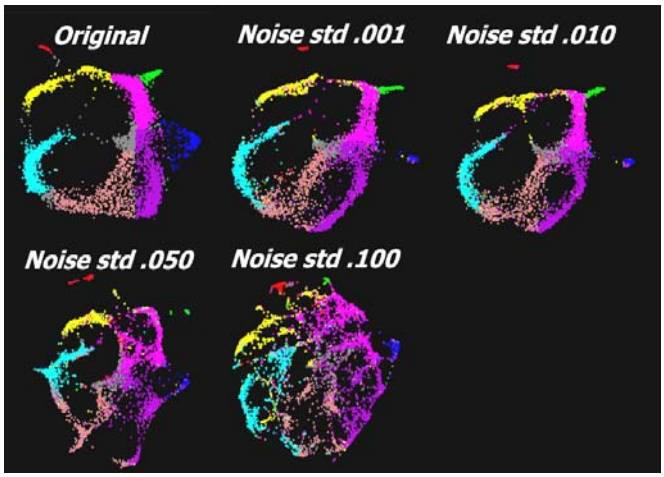
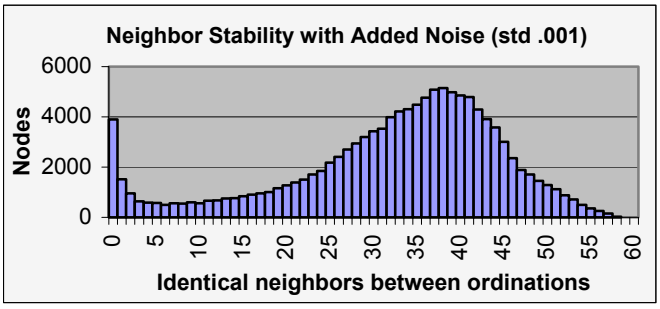


Figure 53. Demonstrates the affect of increasing edge noise on cluster stability.



Figures 54. Histogram of neighborhood stability with added noise (std 0.001).

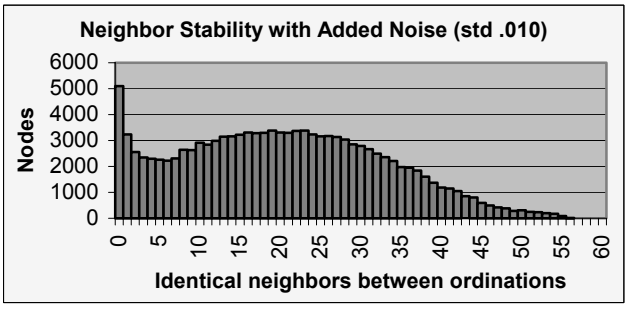


Figure 55. Histogram of neighborhood stability with added noise (std 0.010).

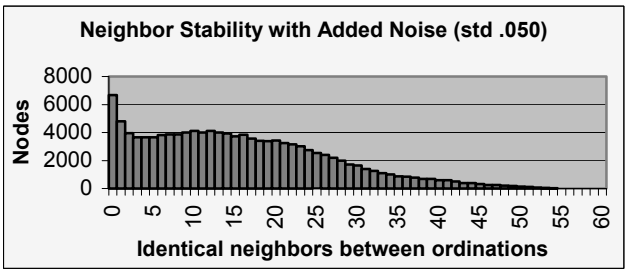


Figure 56: Histogram of neighborhood stability with added noise (std 0.050).

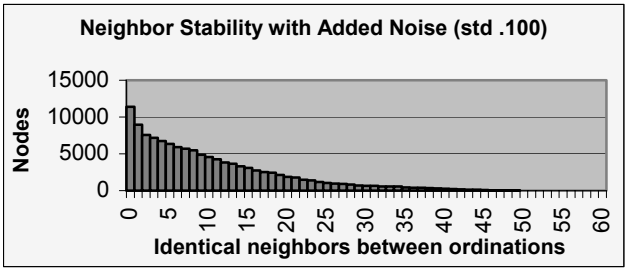


Figure 57 Histogram of neighborhood stability with added noise (std 0.100).

Clustering process discussion

The computational experiments revealed two types of information. First, we discovered that large-scale structures were often very robust to starting with different initial conditions. Second, where there were differences, the insights about why the cluster positions changed were as interesting as the fact that they did change. We present two measures of the stability of these structures: a visual interpretation, and the results of our neighborhood analysis. The visual interpretations are striking in their clarity, but are also supported by the numerical results shown in the histograms.

The histogram numbers can be interpreted as data drawn from a binomial distribution. For example, if the two ordinations were totally random, then the neighbors of a gene in the second ordination would be randomly drawn from all the rest of the genes. Given that we had about 6000 genes, and used a neighborhood size of 60, about 1% of the total genes, the probability of exactly k neighbors in the intersection would be

$$\binom{60}{k}\left(\tfrac{1}{100}\right)^{k}\left(\tfrac{99}{100}\right)^{60-k} .$$

When the size of the neighborhood is 1% of the total number of genes the expected frequency for observing 0 neighbors is about 0.547; the expected frequency for observing 1 neighbor is about 0.332; and the frequency for two neighbors is about 0.099, which leaves the expected frequency for observing three or more neighbors in common to be only 0.022. For 6000 genes, only 132 genes would be expected to have more than two neighbors in common between two random ordinations, which is more while several thousand are actually observed. Hence, the histograms and the visual comparisons show that the differences between pairs of our ordinations are very far from being random.

Figure 52 shows six typical ordinations from different starting conditions. Groups in the first ordination were outlined by hand and colored. These same genes were followed in the other ordinations to observe how their relative positions changed. Two striking patterns emerged. In one case the clusters were almost identical to the initial cluster despite different random seeds. In the second case the resulting clusters are a mirror image of the initial clusters. This mirroring is very reasonable, as there is no reason to expect any preferred natural placement as long as the relative distances are preserved, so rotations and reflections should be, and were observed. The histograms showed good neighborhood agreements between mirrored images.

Closer attention to the structures does reveal a few large changes, for example in Figure 52, where we note that the red cluster has flipped from the inside to an outside configuration. This red cluster has a few strong similarity links tying it to the ridge. As a result, it can easily be mirrored with respect to the ridge. Note that the neighborhood analysis would only detect a few differences along the frontiers of the two clusters. As expected, the histograms show very little difference between the two ordinations with respect to the neighborhood analysis. The most encouraging fact is that most groups not only maintain their relative positions given different starting conditions, but that they maintain similar cluster shapes as well, which indicates good interior agreement, which is, again, supported by the histograms. These results indicate that the ordination tool has robust stability when presented with the same dataset. With that information in hand, we began the investigation of how small changes in the similarity data affected the clustering.

Ideally, one would want an ordination algorithm that responded to slight changes in the similarities by producing slight changes in the ordination and that, in some way, moved smoothly from well ordered groupings to totally unordered, high entropy groupings as the similarities are mixed with more and more noise. Figure 53 shows a starting cluster based on the actual similarities, together with four cases where increasing amounts of noise were added to the correlations. Figures 54-57 are the corresponding histograms reflecting the changes associated with the increasing noise. Note that several large structures remain intact as noise is added, but that some (for example the purple and brown clusters) become more disordered. They essentially melt with increasing noise. Also, note the red and green clusters are apparently more resistant to noise. This melting metaphor is particularly appropriate, because it reflects the internal order that must be 'randomized' or melted before the cluster can begin to break apart.

Mixing increasing amounts of noise with the similarities allows one to quickly see which clusters are more likely to be an artifact; these are the clusters that melt out with the smallest amount of noise. This information is so easy to obtain that we believe it should be part of every analysis based on clustering.

Finding a most representative clustering.

Each randomly restarted ordination by VxOrd represents a sample from a distribution of possible ordinations arising from the particular similarity file. From this perspective, one might want to identify the *best ordination*, which is particularly hard because it is an extreme, and further because the concept of *best cluster or best ordination* is not particularly well defined. However, the idea of a *most representative ordination*, or *most central ordination* (MCO) can be defined with respect to the sample of observed randomly restarted ordinations. In this case, as previously described, two ordinations are compared by neighborhood analysis to create a single measure of overall similarity between the two ordinations. In particular, the sets of the N nearest neighbors are found for every gene in both ordinations. Then the total numbers of intersections are accumulated; that is for every gene, the number of common genes listed in its neighborhood sets are summed together across every individual central gene. The sum may be used directly, or an entropy measure may be computed by weighting the rareness of particular intersection sizes.

With this method for comparing two ordinations, one can make all possible comparisons of the available randomly restarted ordinations and then select the ordination that is, on average, most like all the remaining re-ordinations. This idea of centrality of the distribution of ordinations might be further extended to the second moment to compute some measure of dispersion, which perhaps could eventually be extended to allow some sort of hypothesis testing about these ordinations. However, we have only investigated the centrality issue.

We used massively parallel computers to calculate hundreds, or in some cases thousands, of reclustering ordinations with different seeds for the random number generator. We compared pairs of ordinations by counting, for every gene, the number of common neighbors found in each ordination. Typically, we looked in a region containing the 20 nearest neighbors around each gene, in which case one could find (around each gene) a minimum of 0 common neighbors in the two ordinations, or a maximum of 20 common neighbors. By summing across every one of the genes an overall comparison of similarity of the two ordinations can be computed. We computed all pair wise comparisons between the randomly restarted ordinations and found the ordination that had the largest count of similar neighbors across the totality of all the comparisons. Note that this corresponds to finding the ordination whose comparison with all the others has minimal entropy, and in a general sense represents the most central ordination (MCO) of the entire set. Figures 58, 59, and 60 show the entropies, the intersection counts, and the cross plot of entropy and intersection for a data set that we had, otherwise, found very difficult to break into stable clusters. Even for this experiment, it is obvious that a few ordinations are more central. Note that these results suggest the distribution of possible ordinations is very diverse.

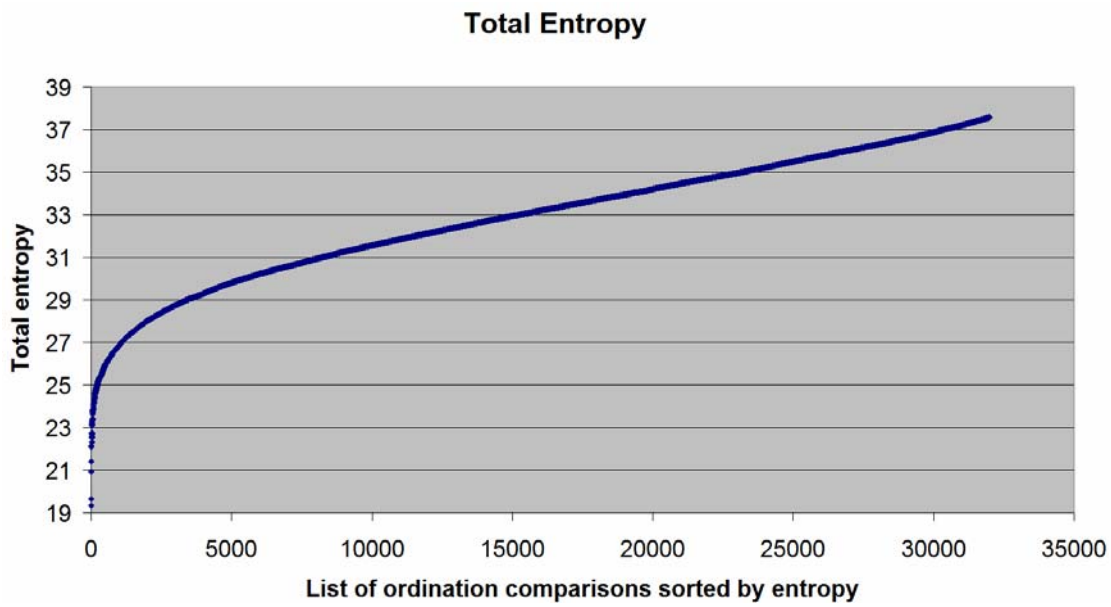


Figure 58. The distribution of observed entropies from the ordination comparisons.

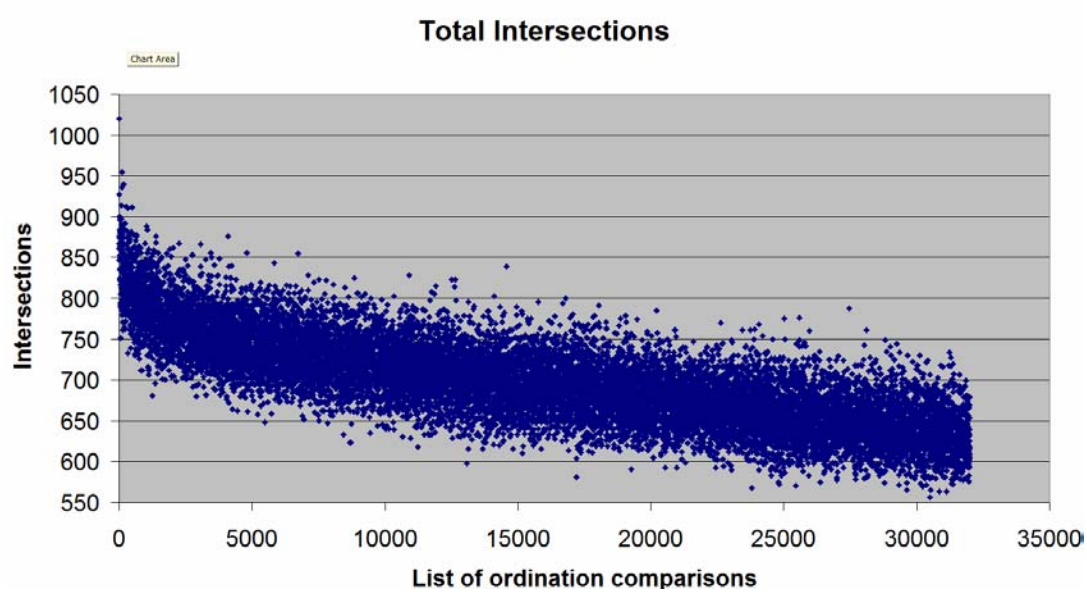


Figure 59. Neighborhood intersection counts from the comparison of ordination pairs sorted into increasing entropy order, as shown in previous figure.

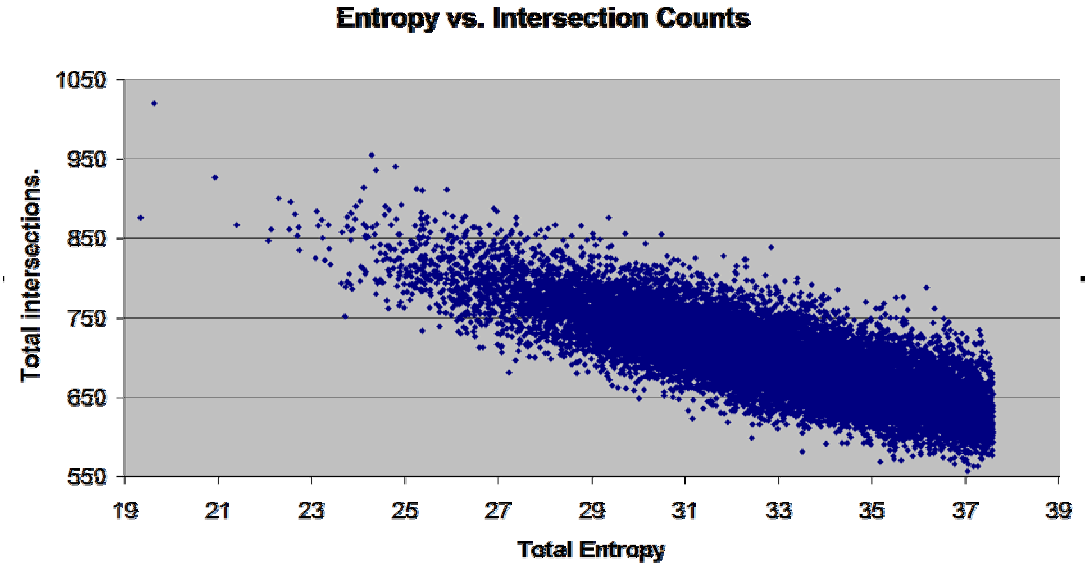


Figure 60. The comparison of total entropy and total intersection counts for the ordination pairs being compared.

It is possible to use these comparison counts (or entropies) *as a derived similarity measure* to compute another round of ordinations. For example, given that 200 random re-ordinations have been computed, one can compute the total number of times gene, G_j, turns up in the neighborhood of gene, G_k, in the available 200 ordinations. This count, or the average number of times the two genes are near each other, will be high when the two genes are generally collocated (which should be a reflection of similar expression profiles for G_j and G_k). The clusters from this recursive use of the ordination algorithm are generally smaller, much tighter, and are generally more stable with respect to random starting conditions than any single ordination. Figure 61 shows the Most Central Ordination from this derived similarity. The locations of the elements of

a few of those clusters are then shown in Figure 62 (by similar coloring) in the Most Central Ordination based on the original similarities. Figure 63 shows the same locations, but within the first of the 200 single re-ordinations, i.e., just one ordination with no attempt at finding centrality. These figures show that the similarities derived by entropies are useful for identifying clusters that may do even better than the Most Central Ordination based on the original similarities from Pearson's R.[7]

We typically use all of these methods (computing the MCO from among about 100 to 200 random re-ordinations, and computing neighborhood set sizes ranging from 10 to 30 by steps of 5) during exploratory data analysis to develop intuition about the data. Interestingly, the process of comparing pairs of ordinations results in an over all similarity between the two ordinations. These similarities can be used to create *clusters of the clusterings*!

Figure 64 show an example where we found that the random re-clusterings seem to fall into two different attractor basins, which may be interpreted as a sign that there are two different, but perhaps equally valuable ways to look at the data, and that no single cluster will be able to represent both of these ways of looking at the data.
This section has attempted to show that clustering is a difficult task, and that stochastic reclusterings should always be examined before accepting any particular clustering. In the next section, we turn to the initial evaluations of a data set using the visualization environment VxInsight. There we will see how VxInsight makes use of a terrain metaphor for the data, which helps an analyst find, and memorize many large scale features in the data. We will also see how it serves as a visual interface to the database of meta-information about the measurements, and as an interface to expression plotting and links to external databases of annotation information.

[7] However, the ordination should be examined for biological sense before deciding which similarity measure is better for a particular experiment.

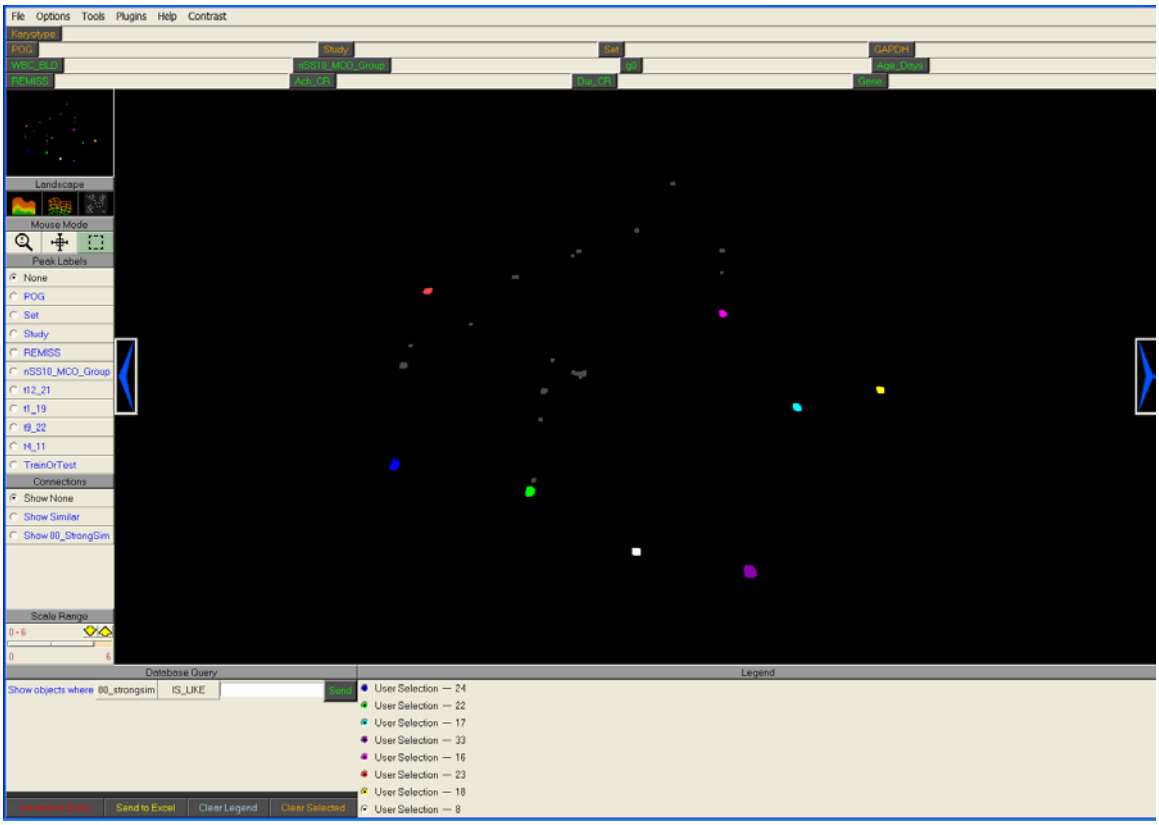


Figure 61. Clusters derived by using the neighborhood intersection counts as a new kind of summary similarity. This type of clustering is often relatively stable, and tightly grouped.

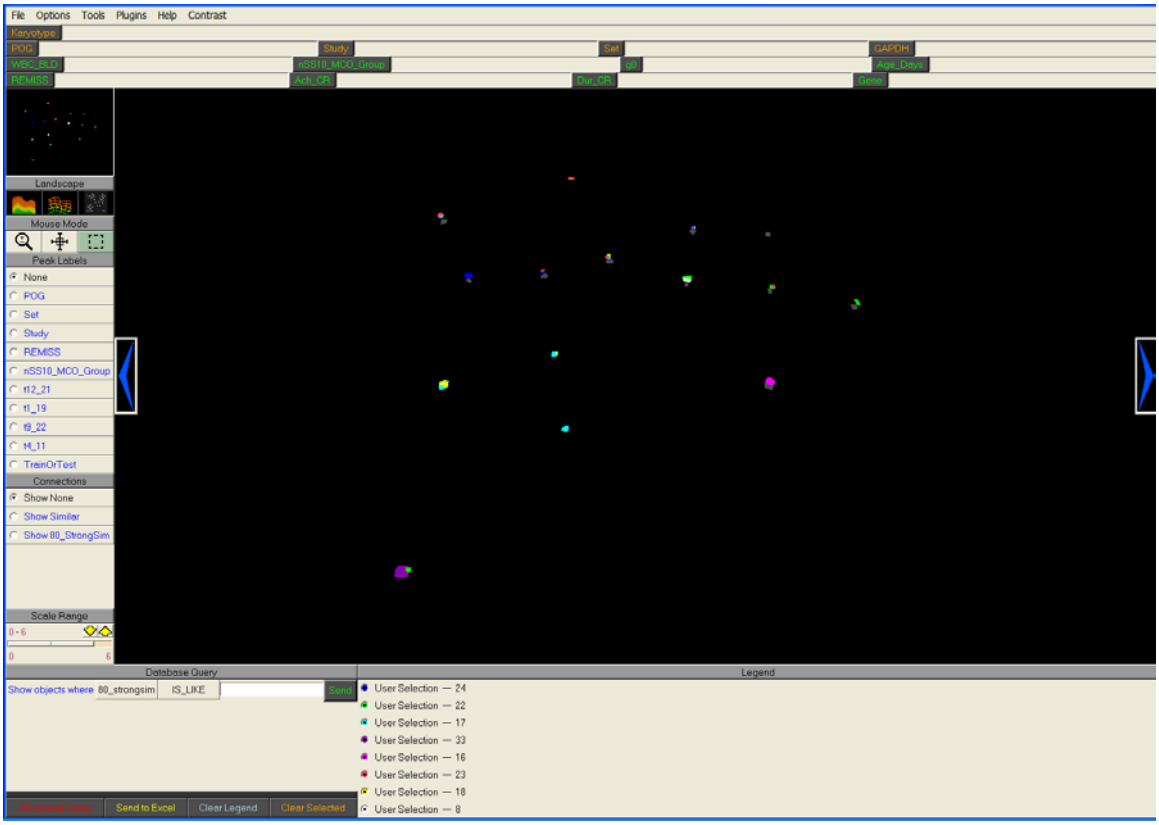


Figure 62. The most central ordination, MCO, created from 200 random ordinations base on the original expression similarities. The colors represent the groupings from in the previous figure, which used a similarity based on neighborhood intersections.

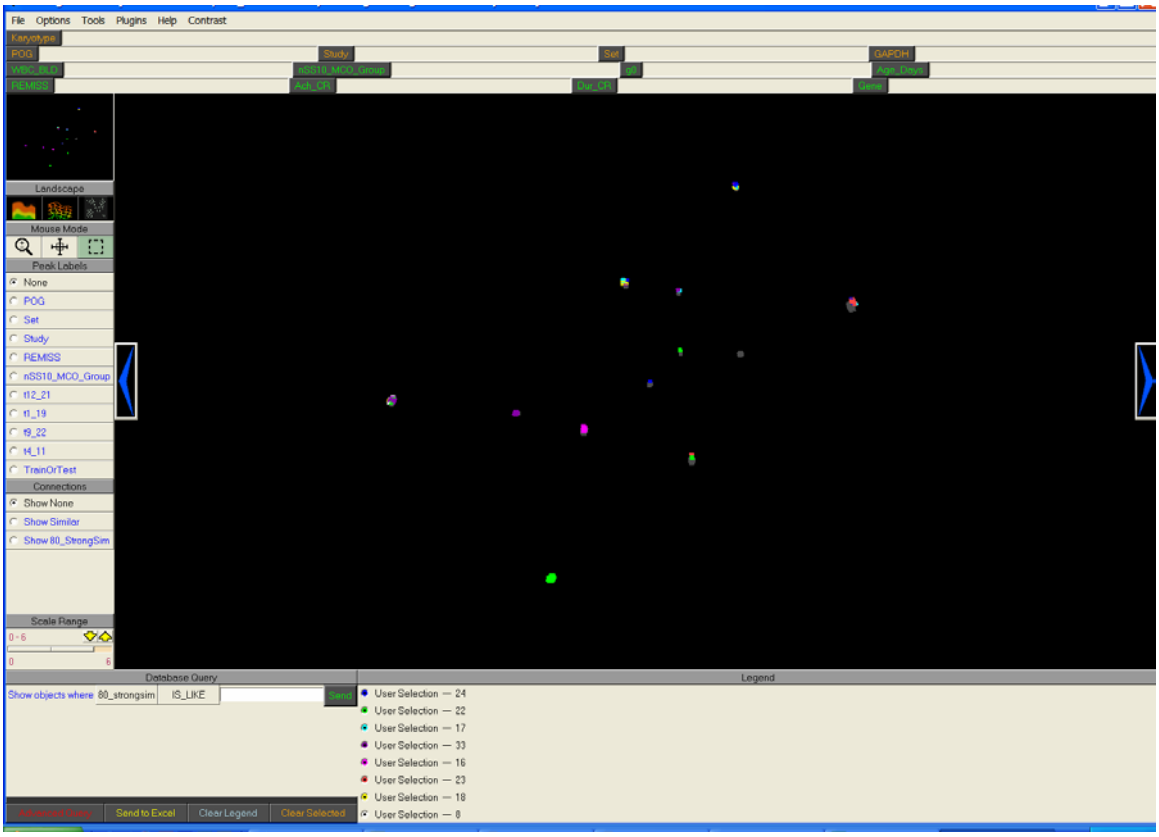


Figure 63. A single ordination based on the original expression similarities. The colors represent the clusters shown in Figure 61 arising from the neighborhood intersection count similarities.

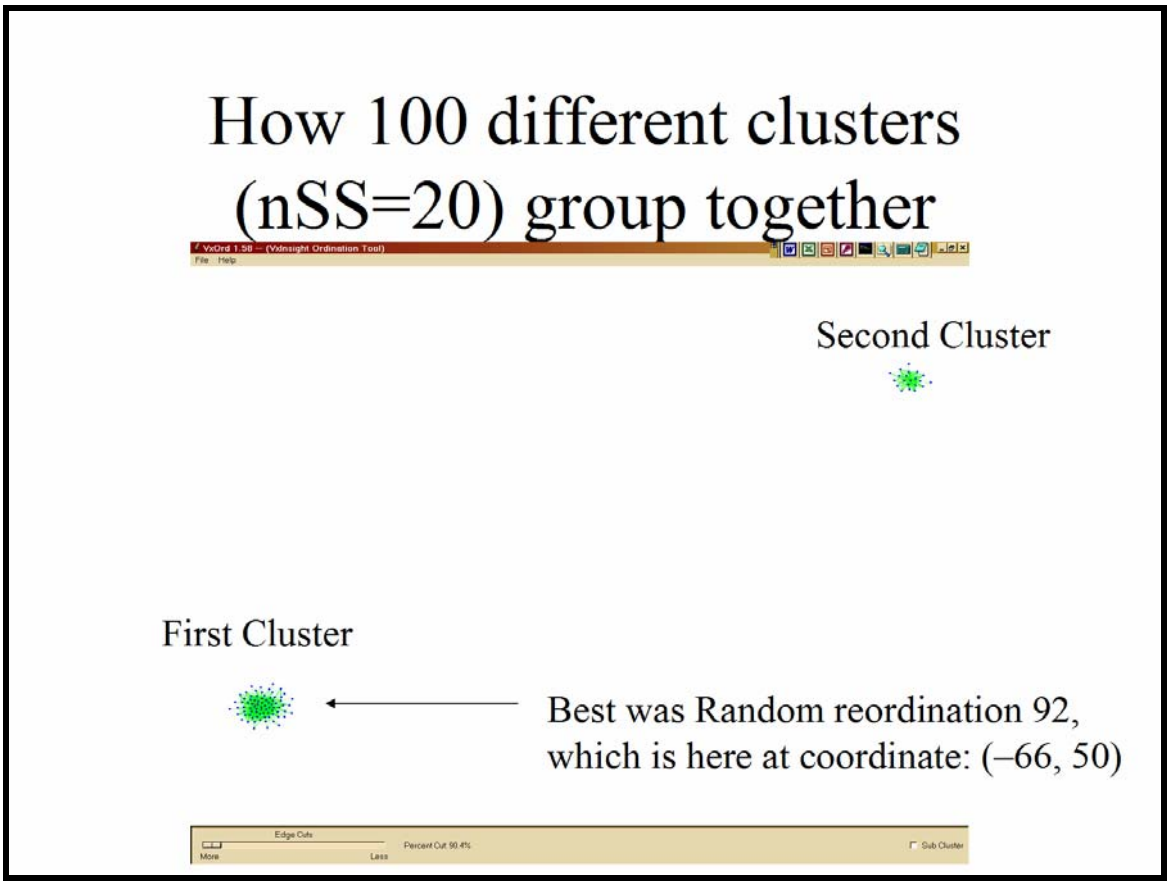


Figure 64. The process of comparing the individual clusters results in an overal similarity between the two clusters. This overall similarity can itself be used to create a cluster-of-clusters. In this particular case, there seem to be two attractor basins, suggesting the data may have two useful projections.

Using VxInsight to analyze microarray data

Microarray experiments often produce such prodigious quantities of data that one is tempted to rely on statistical, or other mathematical filters. These automated approaches should always be evaluated, and exploited to the maximum capacity possible. However, human pattern recognition and anomaly detection can also be very powerful, especially when the large quantities of data are presented in an easily visualized manner, and when the researcher can quickly and easily explore not only the raw data, but connections with external data, such as annotations or clinical information. The VxInsight information visualization and visual, database interface is a very powerful aid to this kind of exploratory data analysis. The previous section discussed how array data can be clustered, or ordinated for use with VxInsight. We now turn our attention to the specific features of this visual interface that are useful for analyzing microarray data.

Figure 65 shows a typical visual presentation of clustered microarray data. The central paradigm is to present the clusters via a terrain metaphor, which has proven to be particularly useful because humans seem to have an innate capability to memorize and navigate through terrains and symbolic representations, or maps, of the terrain. In the case of VxInsight, the height of the

mountain represents the number of data elements clustered under the mountain, and the physical separations encode the relative similarities between the data items, such that mountains closer together will have more similar data elements than mountains further apart. At this highest level one can only see the global structure of the clustering, however, it is possible to zoom into these mountains to see finer and finer structures, all the way down to individual data items, see Figures 65 and 66.

At any time in the analysis, one can form a query to the underlying ODBC-compliant databases, which typically contain specific information such as annotations, or clinical information. These queries are entered using the Graphical User Interface (GUI) in the left hand side of the screen. The results of these queries are displayed not as a list of text, but visually within the context of the clustering (note the query legend at the bottom right hand side of the screen, and the corresponding colored spots across the cluster terrain). Often this contextual presentation is the important result of the query, not the specifics returned from the query.

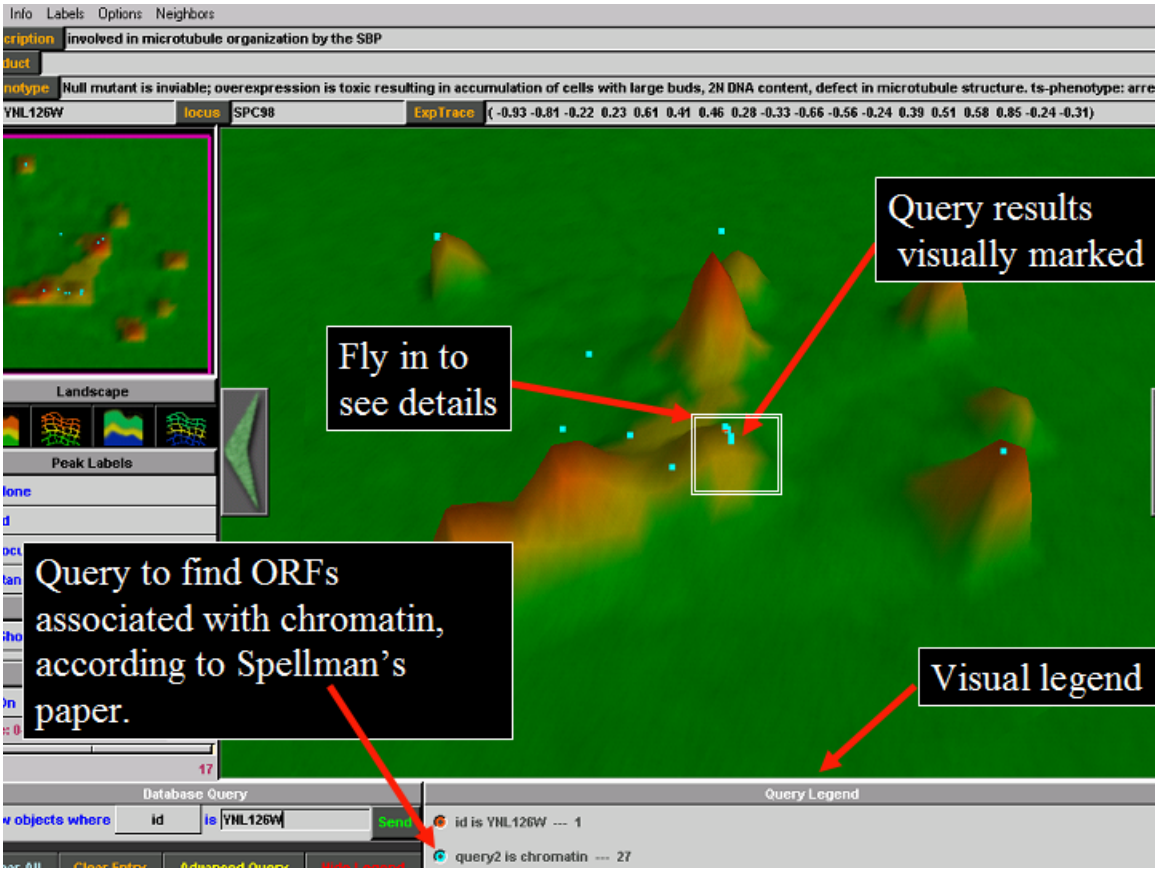


Figure 65. VxInsight terrain metaphor and interaction features.

Figure 66 shows a close up, after zooming into the region in outlined in Figure 65. To help maintain ones sense of direction when viewing the clusters at high levels of magnification, the global location of the current view is displayed in the small image near the upper left hand side of the screen. The magenta square in that image shows the particular location and scale of the larger view, with respect to the overall clustering. This particular image shows the collocation of

a set of genes that were closely clustered together in Eisen's original analysis of the Spellman yeast cycle data.

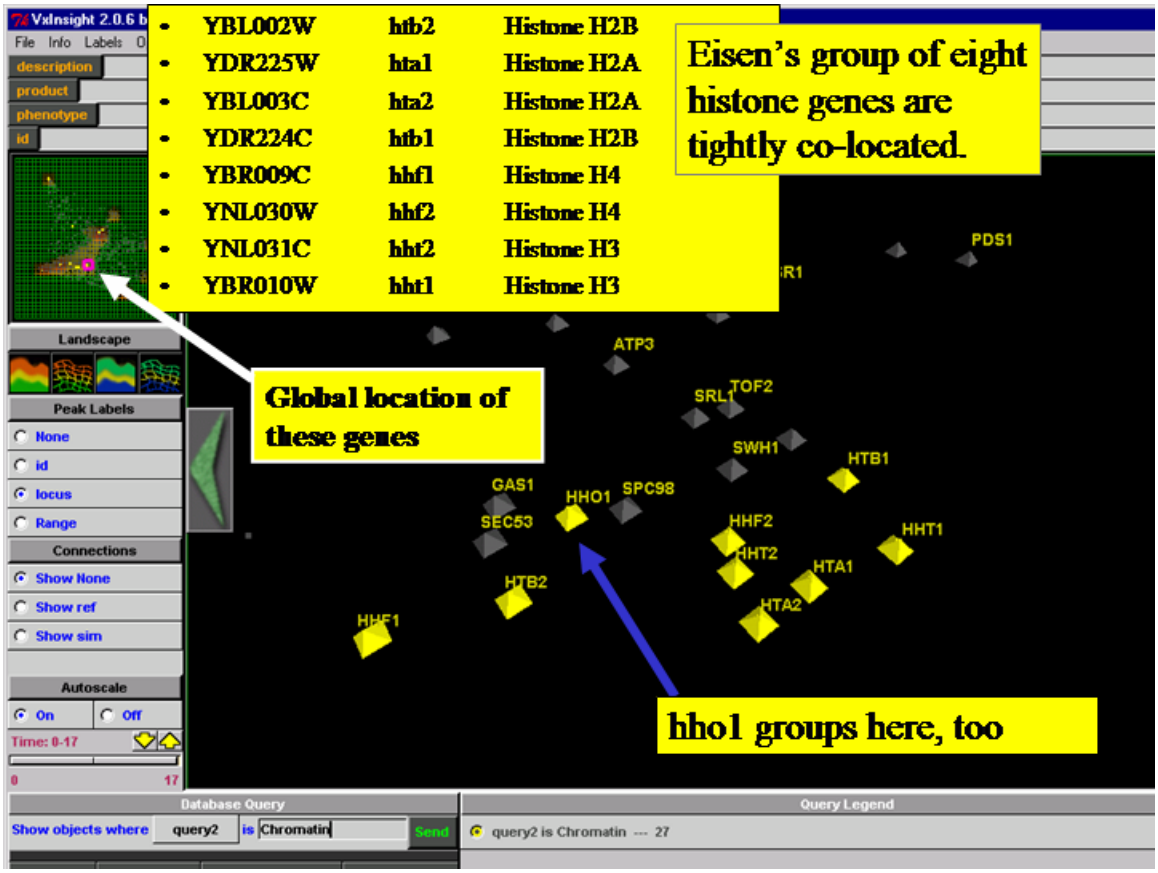


Figure 66. A detail view of the region marked in Figure 65 demonstrating the collocation of histone genes previously known to be closely related.

Figure 67 shows several useful features of the visual interface. For example one can request the display of specific information about the cluster mountains or (at the lowest level) the individual data elements themselves. In this case the gene names are displayed. In other cases, a summary is automatically constructed, based on the collection of elements under the mountains, and displayed above the mountain. Clicking on the representation of a specific element (the individual pyramids) will cause an automatic database query about that element, the results of which are displayed in the form at the top of the screen.

Several analysis features are available from the menu bar at the very top of the screen, for example the expression profiles of a gene and its nearest neighbors can be plotted. Figure 68 shows the popup GUI that allows the control of the number of neighbors to be plotted, and Figure 69 shows an example plot from this cell cycle data set. The produced HTML page is automatically displayed in a browser window, and includes plots and hot links to web-based annotations for each of the genes. These features constitute the general, exploratory analysis tools. We now turn to an example of how these tools have been used in an analysis, and a discussion of a few very extensions to VxInsight which are specific for array analyses.

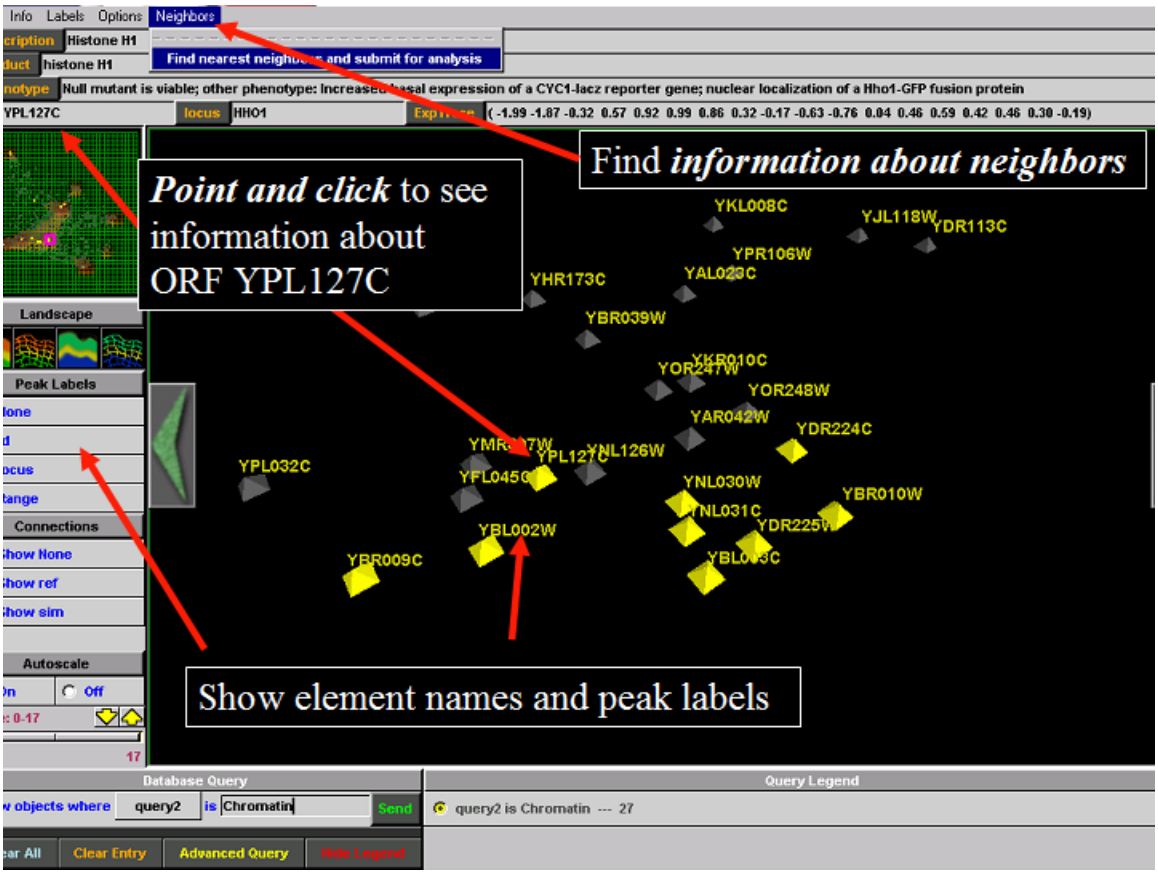


Figure 67. Useful annotation options and features.

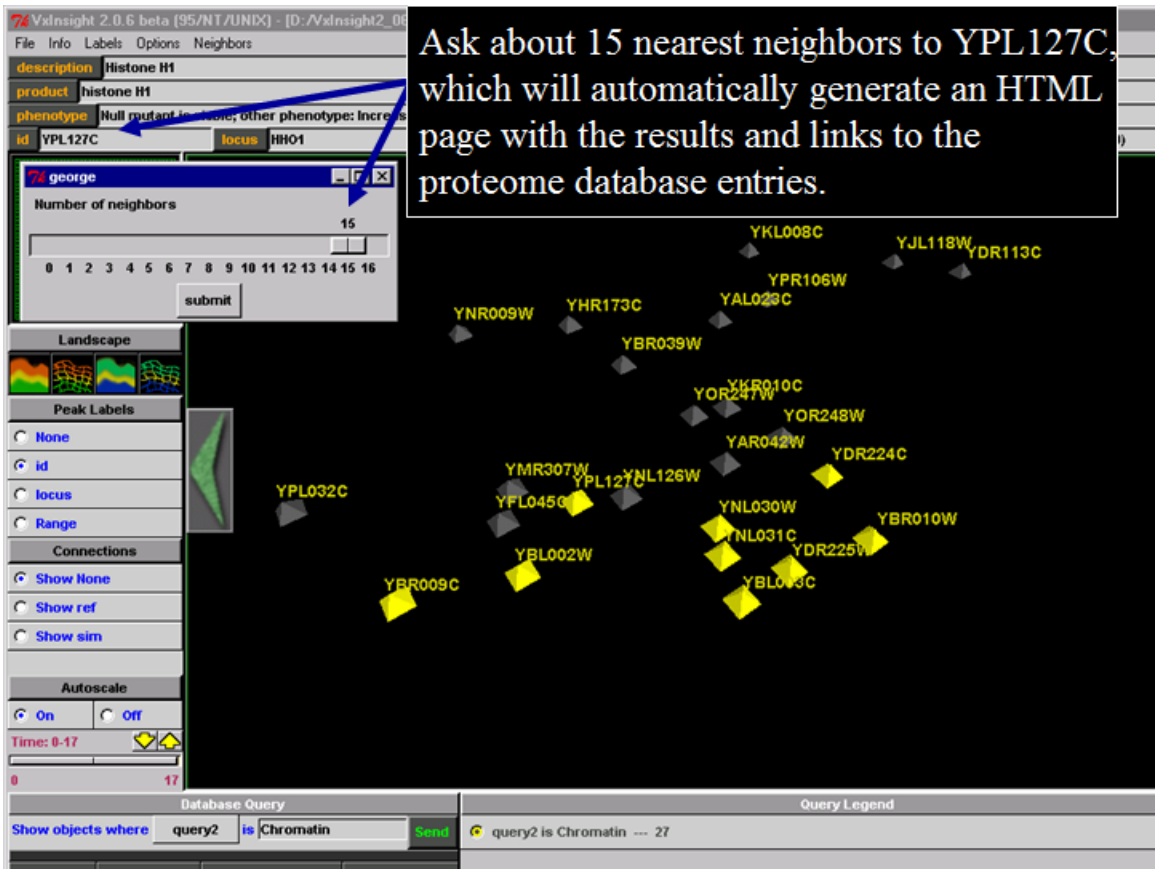


Figure 68. The popup GUI used to request the expression plots for a selected gene and its nearby neighbors.

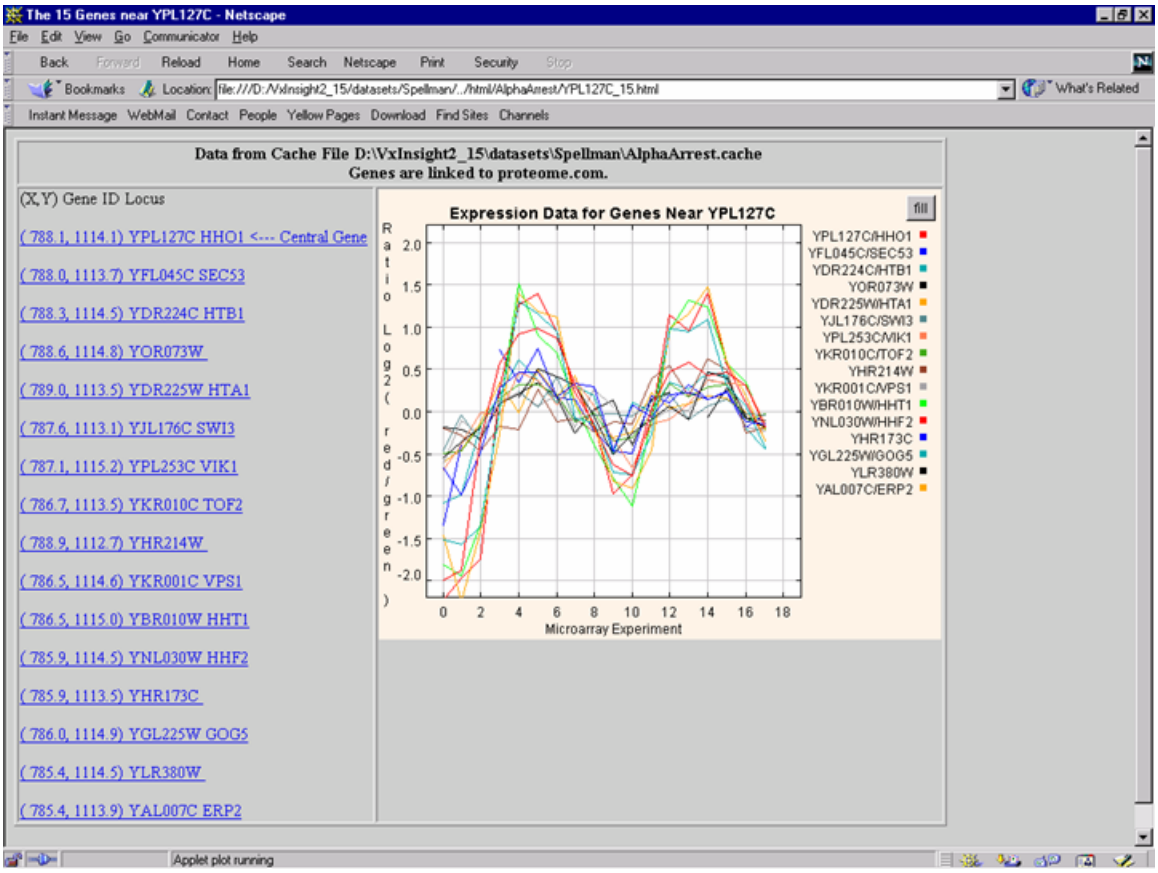


Figure 69. A plot of gene expression levels across two cell cycles, following Figure 68.

A few typical steps in an analysis when using VxInsight

VxInsight is very useful for an initial sanity check of a dataset. We will typically cluster the arrays to look for mistakes in the scanning, or data processing, which might have duplicated an array. A duplication will often be apparent in the experiment because the pair of duplicated arrays will cluster directly on top of each other, and will typically be far from the other clusters. We have discovered that many datasets cluster more by the day the samples were processed, or even by the technician processing the samples, than due to biologically relevant factors. To test this we will use the "label peaks" feature as shown in Figure 70. If almost 100% of a particular processing set clusters by itself, that is a real concern. One can often see the effect of confounding experimental conditions using this same method. For example, if a set of arrays are processed by the date they were collected, and the date corresponds to separate individual studies, then the processing set (date) will be confounded with the study number. Well designed studies control such confounding by randomizing the processing order, or by carefully balancing the processing order. However, it is always wise to use these exploratory analysis methods to ensure that your main effect has not somehow been confounded.

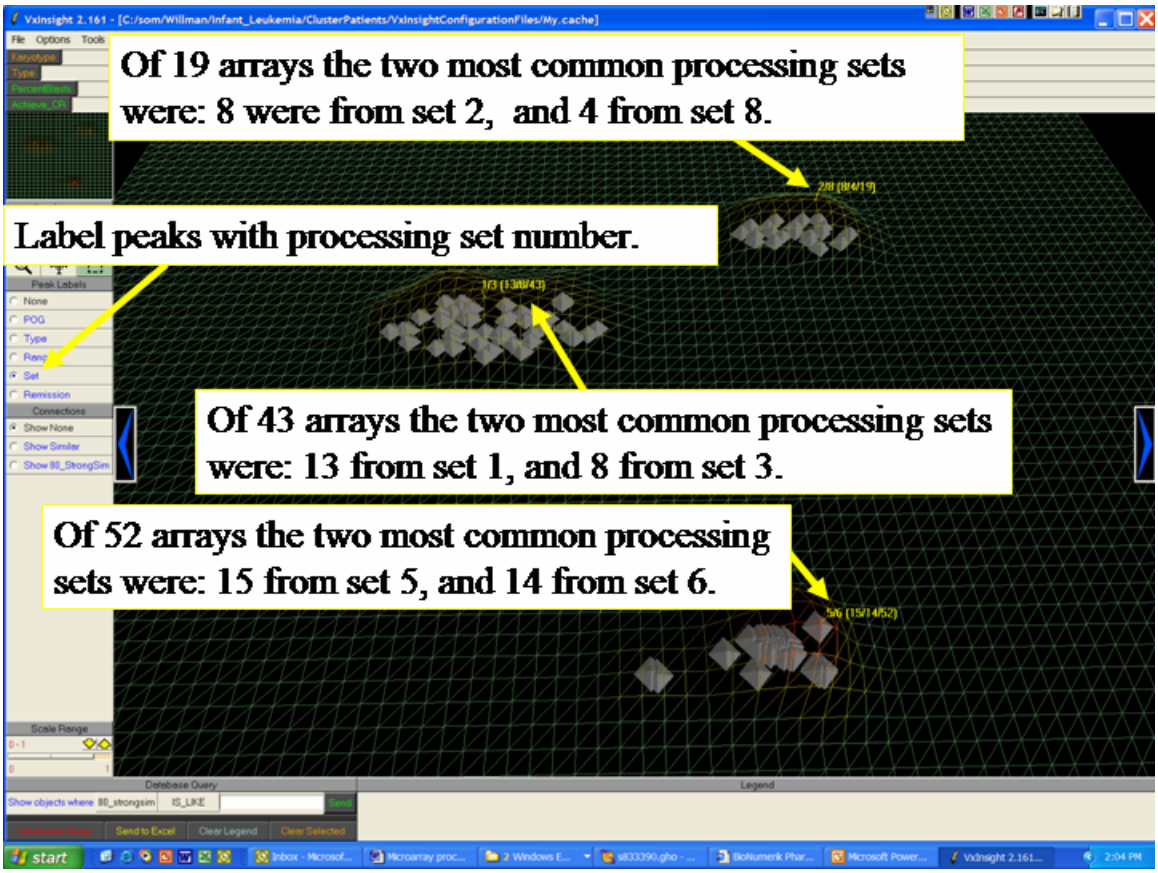


Figure 70. Here the label peaks feature is enabled, which in this case shows the two most numerous processing sets in each cluster. When a cluster is almost 100% from a single set, then the cluster is likely due to set specific conditions rather than biologically important differences.

Figure 71 shows another example, which could be a cause for concern. In this case there are apparently two strong clusters, which have been labeled by an experimental condition. While the experimental condition is not completely driving the separation, there is some separation by that condition within the first cluster (green elements mostly on one side of that cluster and white elements mostly on the other side). However, the two green elements in the midst of all of the white ones in the second cluster are very suspect. Any anomalous event like this should be carefully examined by looking for possible mistakes in processing or labeling. In this particular case, it turned out that all of the arrays in the second cluster were from samples that had extremely poor mRNA yields, possibly due to the age of the samples. The apparently anomalous two green elements were clustered with this group because the expression levels in these samples with disintegrating cells were so different from those samples with healthier cells, which cluster in the left hand group.

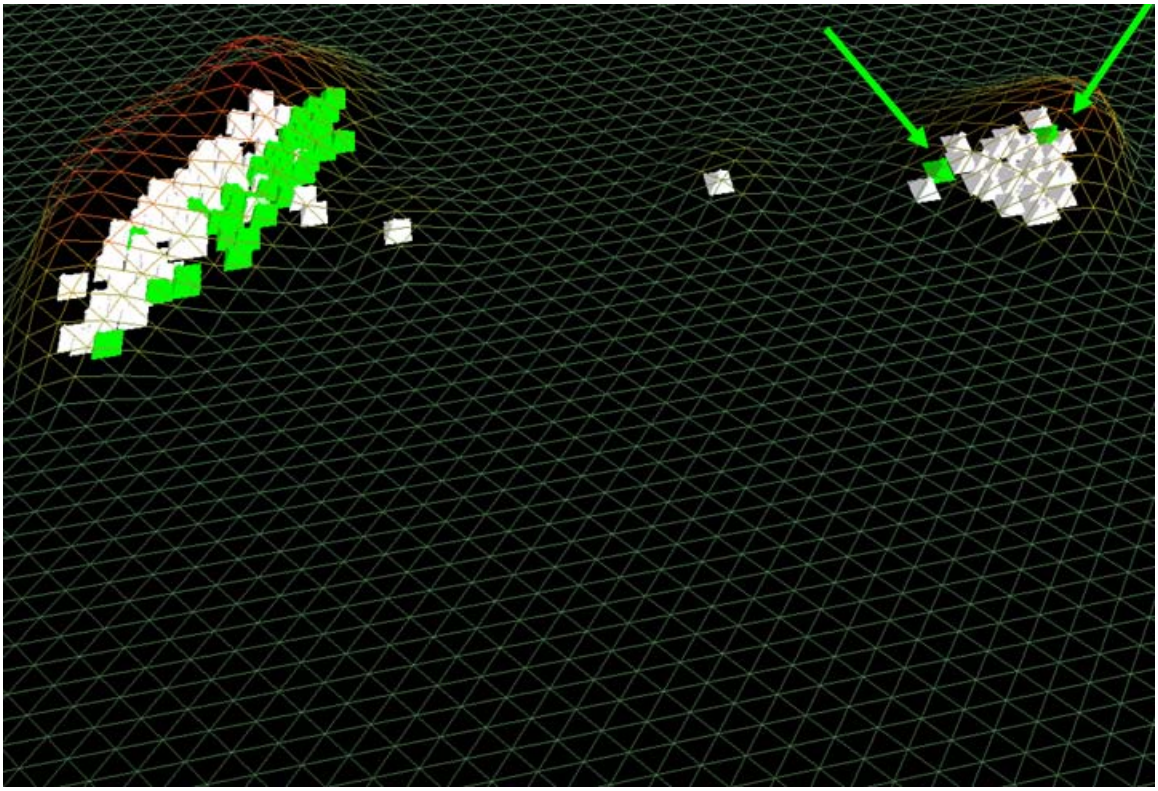


Figure 71. The discovery of two genes (green) clustering in the midst of another large group of genes (white) should raise a flag and lead ought to motivate further investigations to see if these two arrays could have been mislabeled.

A more interesting phase of analysis begins after obviously the bad data have been culled and the remaining data have been reclustered. The data may be clustered in either of two ways. *The genes* may be clustered in an effort to identify possible functions for unstudied genes by using the known functions of genes that are clustered near the unstudied ones (see [33]. and [36] for example).

The other approach, which is often seen in clinical studies, *clusters the arrays* (the patients) by their overall expression patterns. These clusters will hopefully correspond to some important differentiating characteristic; say, something in the clinical information. As the analysis proceeds various hypotheses are created and tested for reasonableness. The plotting features are helpful at this point, especially because the browser page with the plots will also have links to external, web-based information.

Identifying mechanisms which might be responsible for the observed array clusters requires the simultaneously analysis of both the results of clustering by arrays and the original gene expressions. At the highest level, one may wish to select two clusters of arrays and ask *which genes have significantly differential expressions between these two clusters*. Given any method for identifying such genes, it is useful to display them within the context of the cluster-by-genes map. Sometimes the most strongly differentiating genes for the clusters of arrays will not be named, and may not have been previously studied. In this case, it can be very useful to see which known genes cluster around these unstudied genes in that cluster-by-genes map.

This analysis process begins with the use of any one of the available methods to select two or more sets of arrays. Then individual genes are contrasted across the sets with, say, a t-test, or analysis of variance. These statistics can then be used as an index to sort the most highly contrastive genes into an ordered gene expressions. This is schematically described in Figure 72, which shows the original table of array data, which has been clustered both by arrays and by genes. The lower map represents the result after clustering-by-arrays, and shows two highlighted clusters (colored white and green, respectively). The genes with strongly differential expressions between the groups of arrays are shown to the right of this map. Note that the list is sorted by a statistical score, and also contains links to the available web-based annotations. A curved arrow, in the figure, suggests the path between the gene list and the cluster-by-genes image. That connection is implemented with sockets, and forms the basis of a more general analysis tool, which allows an arbitrary gene list to be sent from the analysis of the arrays to the analysis of the genes. The creation and stability of these gene lists is an especially important issue, because they are the first stepping stones toward explanatory stories about possible mechanisms.

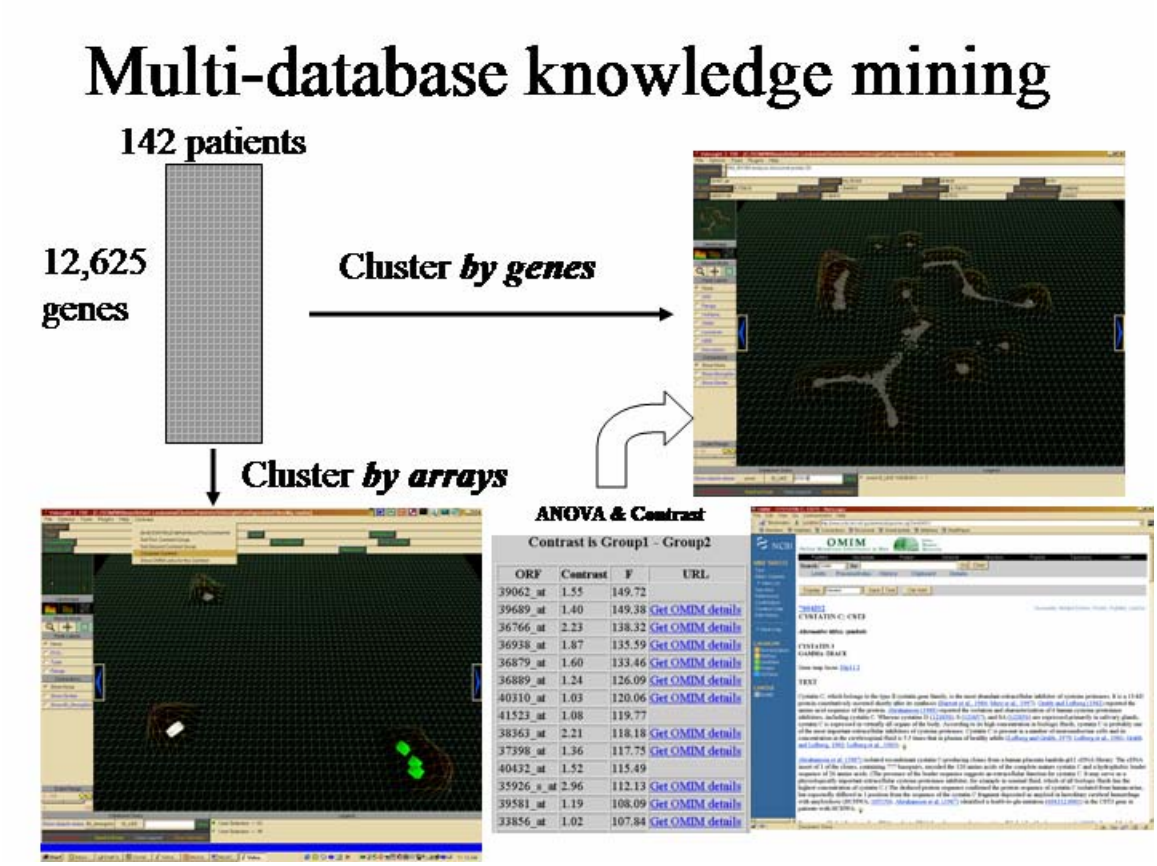


Figure 72. The array of expression data for a large number of experiments is shown being cluster by genes, and also by arrays. A list of genes is shown, which have different expressions between two groups of arrays. This list includes a short annotation, and links to more extensive, web-based annotations.

Generating gene lists and establishing their stability

Analysis of variance (ANOVA) was used to determine which genes had the strongest differences between pairs of patient clusters. These gene lists were sorted into decreasing order based on the resulting F-scores, and were presented in an HTML format with links to the associated OMIM pages, which were manually examined to hypothesize biological differences between the clusters.

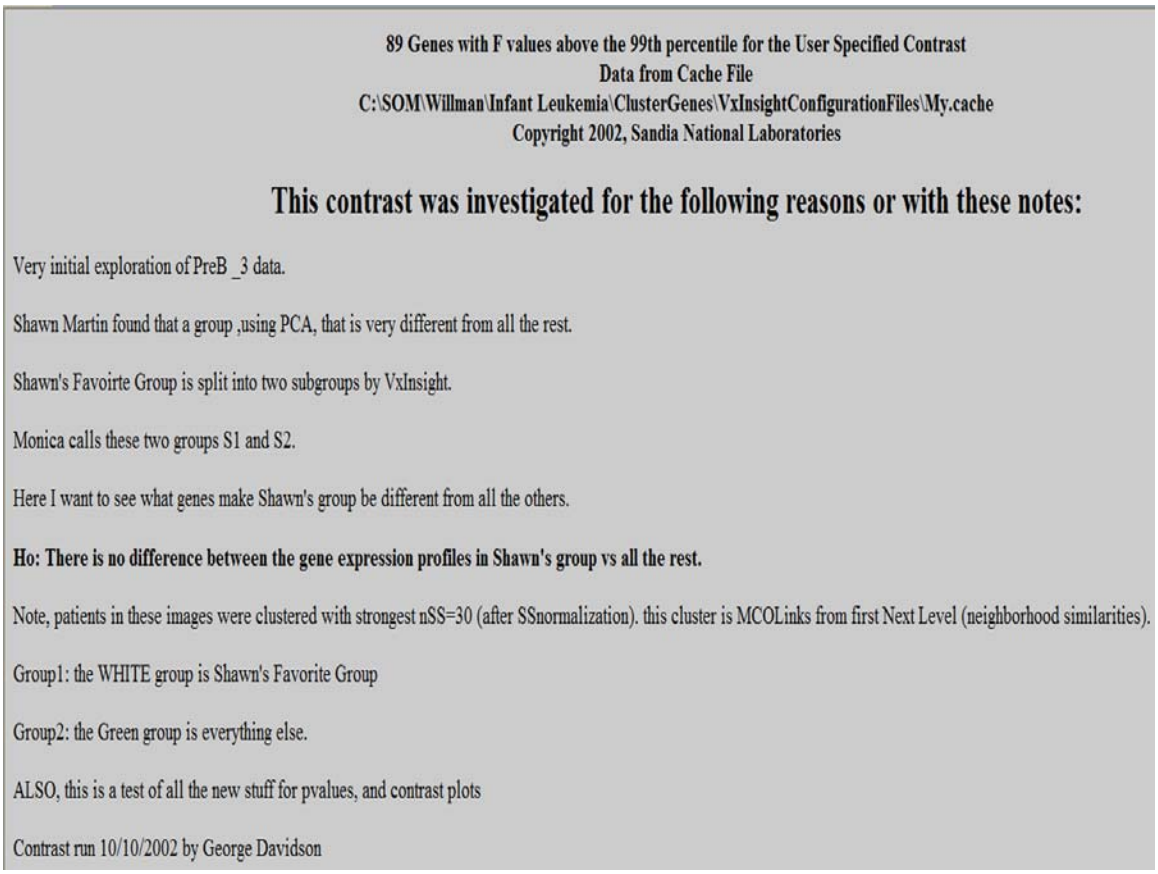
We also investigated the stability of those gene lists using statistical bootstraps (46, 47). For each pair of clusters we computed 1000 random bootstrap cases (resampling with replacement from the observed expressions) and computed the resulting ordered lists of genes using the same ANOVA method as before. The average order in the set of bootstrapped gene lists was computed for all genes, and reported as an indication of rank order stability (the percentile from the bootstraps estimates a p-value for observing a gene at or above the list order observed using the original experimental values).

Identifying gene lists

A list of genes with differential expression can be found for questions such as, "which genes differentiate Cluster-1 from Cluster-2," or "which genes differentiate Cancer-1 from Cancer-2." In the first case the selection of the groups can easily be done by drawing rubber band lines around the clusters inside the VxInsight display. In the second case, the groups are most easily defined by means of a query to the associated database. In either case, once the groups of arrays are available, the entire set of genes can be searched to find those which have the greatest differences in expressions between the two groups. There is a wide variety of approaches to finding such an ordered list. Here, we use very straightforward statistical techniques.

A gene-by-gene comparison between two groups, Group-1 and Group-2, can be accomplished with a simple t-test, however, we wanted to eventually support comparisons between more than two groups at a time, so we actually use Analysis of Variance (ANOVA). This processing results in an F-statistic for each gene. The list of genes is sorted to have decreasing F-scores, and then the top 0.01% of the entire list is reported in a web-page format. Figures 73, 74 and 75 show the three sections of the HTML report.

We have found that it is important to capture a minimal amount of information about why the analysts wanted to run the contrast. Without this information, it is very easy to accumulate many pages of information, but after a few weeks no one may be able to remember the details about the specific query, nor the motivation for making it. As a result, the analyst first enters a few lines of free text, which is placed at the beginning of the HTML file, see Figure 73. Typically, this information includes who ran the contrast, the date it was run, information about the way the data had been previously processed, and most importantly, the reason for the contrast including the hypotheses being tested. For the same archival reasons, the VxInsight screen images are captured to show the locations of the selected arrays, i.e., where they are included in the two groups, see Figure 74. Finally, Figure 75 shows the first few genes in the differentiating genelist. A plot of the contrasted expressions may be obtained by clicking on the contrast (the mean expression in Group-1 minus the mean expression in Group-2) value for a gene, see Figure 76. Figure 77 shows the NCBI webpage with the OMIM annotations for the gene top gene, which may be accessed from the genelist page by selecting the OMIM Details link.



89 Genes with F values above the 99th percentile for the User Specified Contrast
Data from Cache File
C:\SOM\Willman\Infant Leukemia\ClusterGenes\VxInsightConfigurationFiles\My.cache

This contrast was investigated for the following reasons or with these notes:

Very initial exploration of PreB _3 data.

Shawn Martin found that a group ,using PCA, that is very different from all the rest.

Shawn's Favoirte Group is split into two subgroups by VxInsight.

Monica calls these two groups S1 and S2.

Here I want to see what genes make Shawn's group be different from all the others.

Ho: There is no difference between the gene expression profiles in Shawn's group vs all the rest.

Note, patients in these images were clustered with strongest nSS=30 (after SSnormalization). this cluster is MCOLinks from first Next Level (neighborhood similarities).

Group1: the WHITE group is Shawn's Favorite Group

Group2: the Green group is everything else.

ALSO, this is a test of all the new stuff for pvalues, and contrast plots

Contrast run 10/10/2002 by George Davidson

Figure 73. The top portion of the web page records the reasons that motivated the query, the hypotheses being tested, and details about the data and the scientist who made the query.

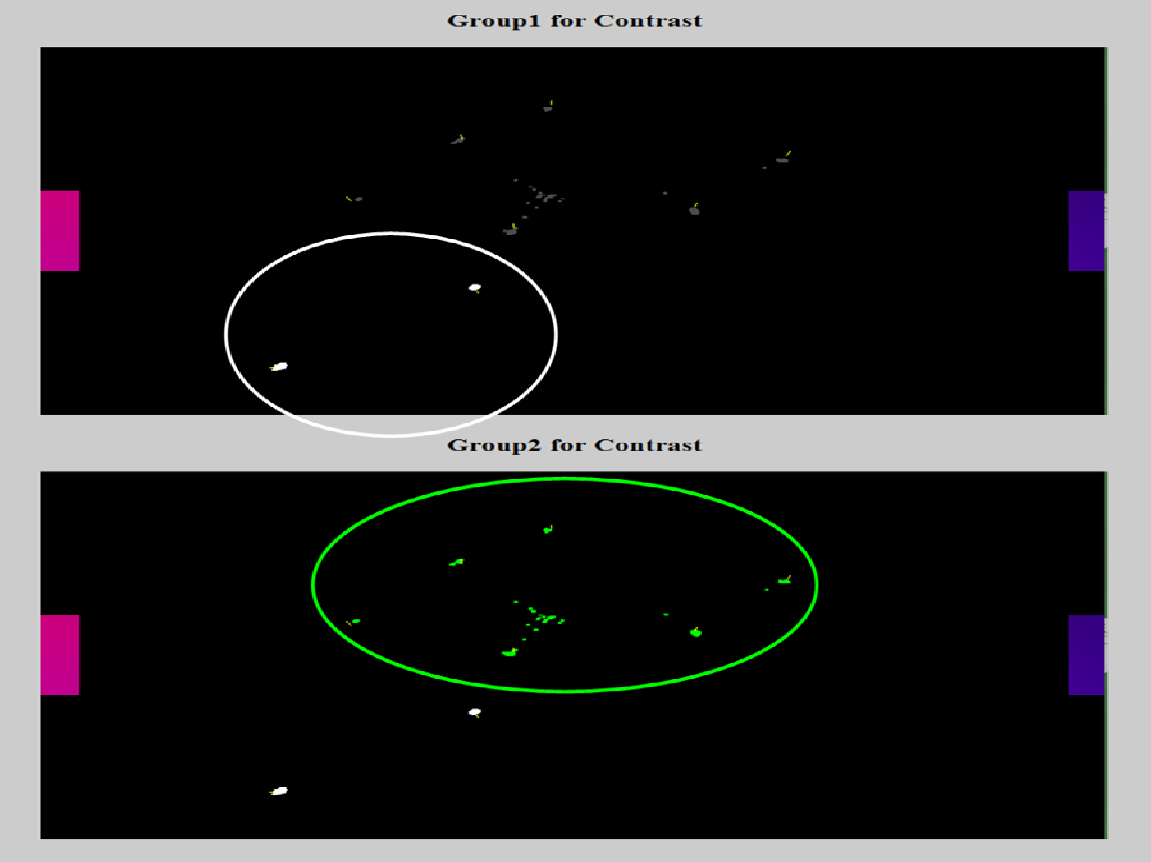


Figure 74. The second part of the web page shows the locations of the two groups being contrasted.

Contrast is Group1 - Group2

Order	ANOVA_F	ORF	Contrast	Bootstrap avg. order*	OMIM	Description
1	2797.09	38319_at	11850.82	[1<=[1<=**1.0** {<=1} <=1] p<=0.001	Details	NM_000732 analysis CD3D antigen delta polypeptide TiT3 complex
2	927.47	38147_at	5329.98	[1<=[2<=**2.3** {<=3} <=3] p<=0.001	Details	NM_002351 analysis SH2 domain protein 1A
3	739.70	39226_at	4924.31	[1<=[2<=**3.4** {<=7} <=7] p<=0.001	Details	NM_000073 analysis CD3G gamma precursor
4	592.20	33238_at	9579.92	[1<=[2<=**5.0** {<=9} <=11] p<=0.001		
5	538.43	2059_s_at	9179.34	[1<=[3<=**5.8** {<=12} <=12] p<=0.001	Details	NM_005356 analysis lymphocyte-specific protein tyrosine kinase

Figure 75. The final part of the web page showing the ordered gene list together with details about its stability and links to the data plots and more detailed annotations at OMIM.

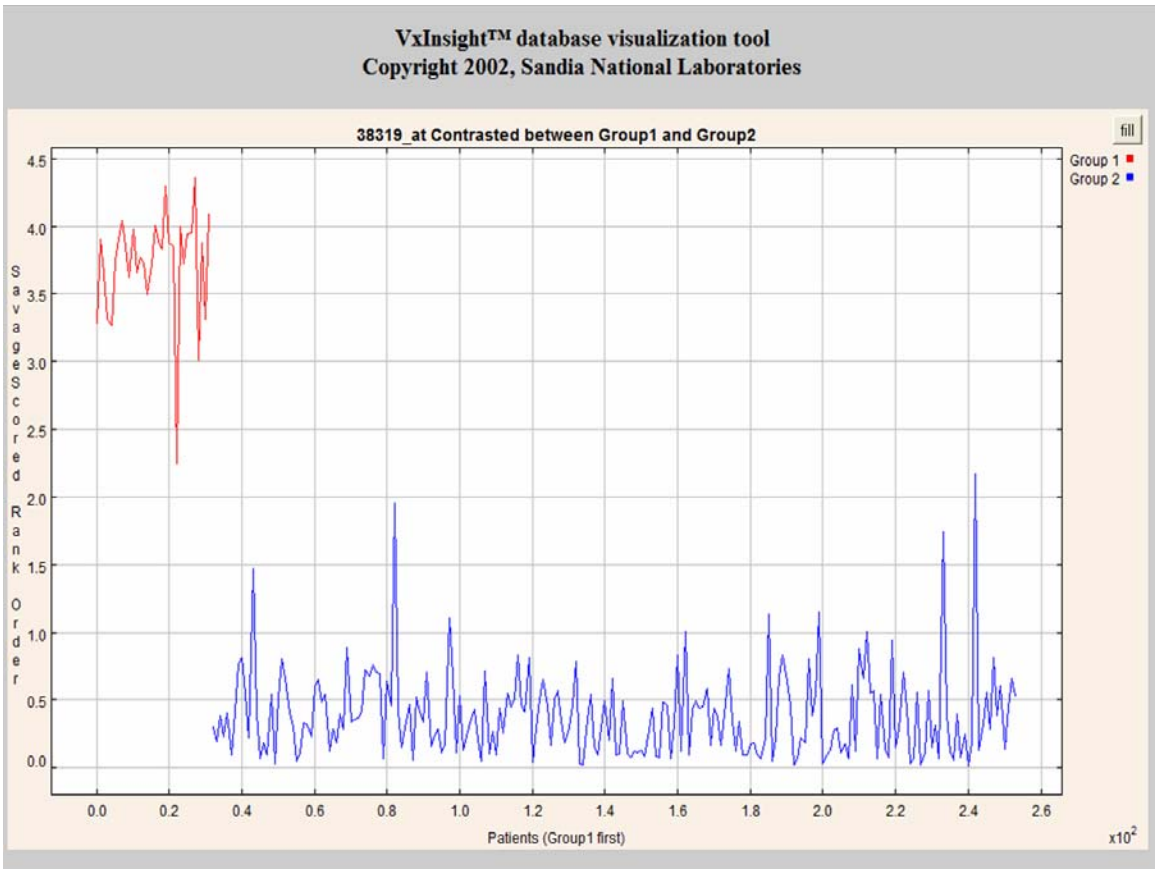


Figure 76. A typical expression plot showing the expression levels for the two groups.

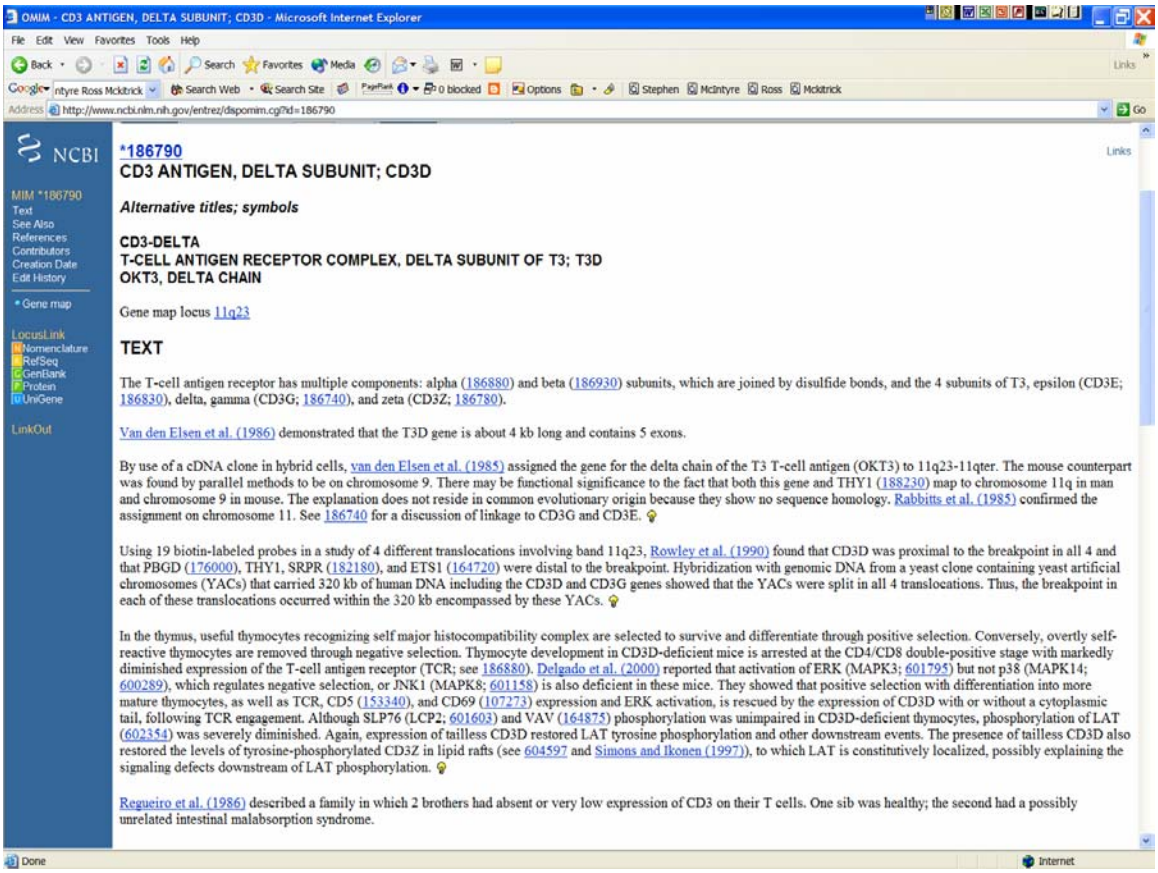


Figure 77. The gene annotations at OMIM are accessable directly from the genelist. These are often sufficient to understand the function of the gene, but references to the primary literature are also accessible from this web page.

These figures indicate how an analysis typically progresses. First a question is posed within the VxInsight framework and a statistical contrast is computed for that question. The gene list is initially examined to see if any genes are recognized by their short descriptions, which, if available, are included with the genes. The plots are examined, and the OMIM annotations are read. If the gene appears to be important, the literature links and other relevant NCBI resources are studied. This analysis step is very labor and knowledge intensive; it requires the bulk of the time needed to make an analysis. As such, it is very important to not waste time following leads that are only weakly indicated. That is to say, before one invests a great deal of time studying the top genes on a list, it is important to know that those highly ranked genes would likely remain highly ranked if the experiment could be repeated, or if slight variations, or perturbations of the data had occurred. The column labeled "Bootstrap average order" in Figure 75 encodes the upper 95% confidence band, the centered 95% confidence band, the average rank order of the gene as derived from the bootstrap computations discussed below.[8] A gene that is not consistently ranked

[8] For example in Figure 76, the gene NM_005356, analysis lymphocyte-specific protein tyrosine kinase, was observed to be the fifth ranked gene with the actual data, while it was found to have an average rank order of 5.8 across the bootstraps. Further, 95% of the time that gene was ranked at or above rank order 12, which is the one-sided upper confidence band for the ranking. Also, 95% of the time it was ranked between 3 and 12, which is the centered confidence band for its ranking. Note, that the lower ranking for both confidence bands is reported to be 12, which is

near the top of the list is probably not one that should be investigated in detail. That column also includes a p-value, which indicates the fraction of time this gene was ranked at its observed (or higher) list position given the assumption that there was no true difference between its expression in Group1 and Group2, which was, again, computed by bootstraps as described below.

The critical issue about any ordered list of genes is, "can we have any confidence that this list reflects any non-random trend?"[9] To be very concrete, suppose that My Favorite Gene (MFG) is at the top of the list from our ANOVA calculations, i.e., MFG had the largest observed F-statistic from the ANOVA. What can we conclude about the observed ranking for MFG? Certainly, a naive use of the F-statistic has no support because we tested, say, 10,000 genes and found the very largest statistic from all of those tests. So, an F-value for p=0.001 would likely be exceeded about 10 times in our process even if all the numbers were random. Hence, the reported F-statistic should only be considered to be an index for ordering the values.

However, if we could repeat the experiment, and if MFG was truly important, it should, on average, sort into order somewhere near the top of the gene list. We cannot actually repeat the experiment, but we can treat the values collected for a gene as a representative empirical distribution for the respective groups. If we accept that this distribution is representative, then we can draw a new set of values for each of the two groups by re-sampling the corresponding empirical distributions repeatedly with replacement, see Figure 78 for a schematic representation of these distributions. This is Efron's bootstrapping insight[48], and forms the basis for our processing.

---

possible when a ranking of 12 is observed many times so that the upper confidence band includes only some few of the rankings of 12th in the list, while the centered confidence band will naturally include more of them.

[9] Every list of, even random, numbers can equally well be sorted into order and will, of course, always have a top ranked value.

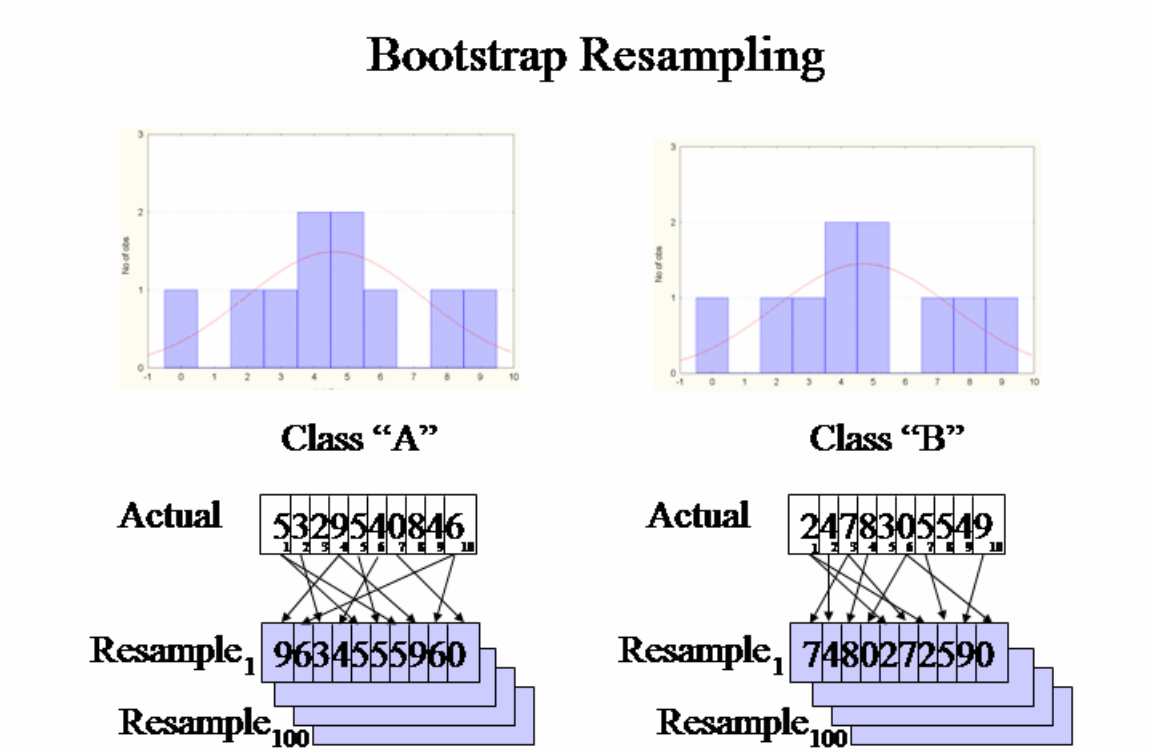


Figure 78. A bootstrap method uses the actual measured data as an estimate for the underlying distribute from which that data was drawn. One can then sample from that estimated underlying distribution by resampling (with replacement) from the actual measurement.

Consider Figure 79, where we resample for every gene across all of the arrays in the two groups to create, say, 100 new experiments that are then processed exactly the same way as the original measurements were processed. We compute ANOVA for each gene and then sort the genes by their F-value. As we construct these bootstrapped experiments we accumulate the distribution for where in the list each gene is likely to appear. Using these bootstrap results one can determine, for each gene, its average order in the gene lists. The distributions for such order statistics can be written, but they are complex. On the other hand the bootstrapped distributions are easily accumulated and are acceptable for our needs. In addition to the average ranking, we count the 95% confidence bands for each gene's ranking as estimated by the bootstraps. We report both the upper 95% confidence band and the 95% confidence interval centered around the mean ranking for each of the genes. The lower limit of this upper 95% confidence band, LLUCB, is recorded for later use (note that 5% of the time we would observe a ranking below LLUCB by random chance even when H_o is false, given the two empirical distributions).

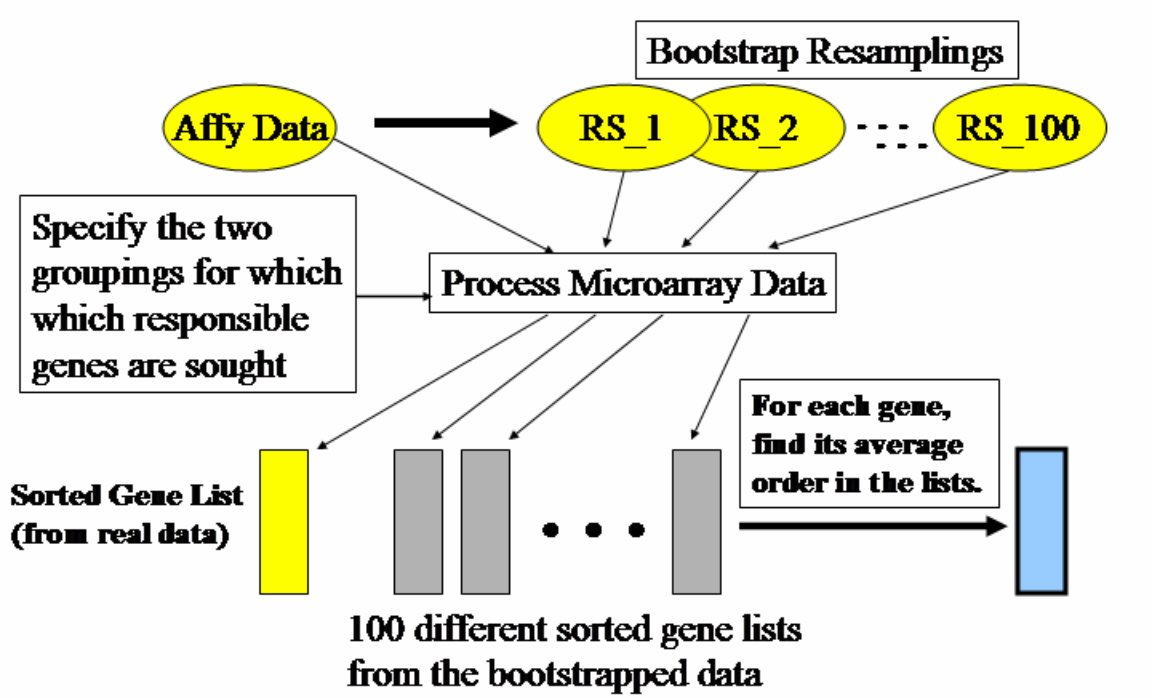


Figure 79. The actual data is processed to create the genelist schematically shown at the bottom left. Then the actual data is resampled to create several bootstrapped datasets, which are processed exactly the same way as the real data to produce a set of genelists. The average order, and the confidence bands for that order, can be estimated from this ensemble of bootstrapped genelists.

A caveat is warranted for these confidence bands. They do not imply that we have a p-value of 0.05 for a ranking, but they do strongly suggest how important a gene is in separating the arrays into the two groups. For instance, if a gene really has no power to separate the groups, then on the average we would see it have a very low average rank order. On the other hand, if a gene is consistently near the top of each of the bootstrapped gene lists, then that gene may be worthy of further investigation.

We can, however, investigate the p-values for the observed rankings of these genes under the null hypothesis, that there is no difference in gene expression between the two groups (Group-1 and Group-2). In this case (*when H_o is in fact true*) *the best empirical distribution would be the unordered combination of all the values without respect to their group labels*. To test this hypothesis, we create, say, 10,000 synthetic distributions by bootstrapping from this combined empirical distribution, and process them exactly as we did the original data.

We are interested in what fraction of the time we observed a particular gene ranking higher in the bootstrapped results than the appropriate critical value. There are several reasonable choices for this critical value. We could use the actual observed ranking, or the average ranking from the bootstraps under the assumption that H_o was false. Instead, we take an even more conservative stance and choose a critical value using a power analysis to control our chance of a Type II error, we set beta=0.05, or 5%.

If H_o were false (i.e., if the groups do have different means) then the earlier bootstrapping experiments suggest that one might randomly observe a ranking as low as LLUCB about 5% of the time. Hence, we examine the later bootstrap experiments (under H_o assumed true, and thus no group differences) and find the fraction of the times that we observe a ranking at or above LLUCB. This value is reported, gene-by-gene, as the p-value for the actual rankings. In essence, we are saying that if H_o is true, then by random chance we would have seen the gene ranking above LLUCB with probability p. *As LLUCB is much lower than the actual ranking, this p-value is very conservative for the actual ranking*.

To investigate the meaning of the actual F-statistics used to index these gene lists, we computed another bootstrap experiment. We were interested in the effect of scaling the original expression values by their savage-scored order statistics. As previously discussed, this scoring is felt to be more robust than taking logs. However, we were concerned that this might influence our p-values, so we developed a code to estimate the expected F-statistic for the n-th ranked gene in a gene list from two groups, Group-1 and Group-2 respectively having j and k arrays. This code computes a large bootstrap after randomizing the savage scores within each of the j+k arrays. The code then computes the ANOVA for each gene and eventually sorts the resulting genes into decreasing order by F-statistics. The final result is a p-value (by bootstrap) for the two groups with the specific number of arrays. This computation is rather intensive, and should either be fully tabulated or run only as needed for genes uncovered by the earlier methods. We have not run extensive simulations of this code against the p-values or the list order distributions, but the limited checks did suggest that genes which always ranked near the top of the differentiating gene lists do have rare F-statistics based on the savage-scored orders relative to the expected random distributions (data not shown).

Comparing gene lists

The ANOVA plus bootstrap approach described above is only one way to find genes which may have important roles with respect to particular biological questions. For example, others have produced gene lists by principal component analysis (PCA), gene shaving, Bayesian networks, various forms of machine learning, fuzzy sets, and other classical statistical methods, among many other methods. By using several of these methods one might hope to find a consensus list of genes. Our experience has show that this is possible; while the lists from different methods are generally not exactly the same, they often do have large intersections. However, the simultaneous comparison of multiple lists has been a difficult problem.

We have developed a number of methods to help us understand that the lists may be different in the details, but still very similar biologically, which makes sense considering that different methods might identify different, but closely related elements of regulation or interaction networks. In that case, the methods are suggesting the importance of the network, and the particular region in that network, even though they do not identify exactly the same elements. This relatedness suggests something similar to the kind of "guilty by association" method that has been used to impute gene functions to unstudied genes that cluster near others with known function, as in[33]. Indeed, something similar can be used to evaluate the similarity of multiple gene lists.

Figure 80 shows a VxInsight screen for clusters of genes. Highlighted across the clusters are genes identified by different methods (shown in different colors). In this particular case one can see that the various methods do identify genes that are generally collocated, which suggests that gene regulations and interacting networks do probably play a strong role with respect to the question under consideration (here, for example, the question was "which genes are differentially expressed in two types of cancers."). However, multiple methods do not always produce such strong agreement, as shown in Figure 81. In this case the question was, "which genes are predictive for patients who will ultimately have successful treatment outcomes," and no clear consensus is apparent. Interestingly, the ANOVA plus bootstrap method suggests a very stable set of genes for the first question, while the list for the second question is not stable and has confidence bands spanning hundreds of rank order positions (data not shown).

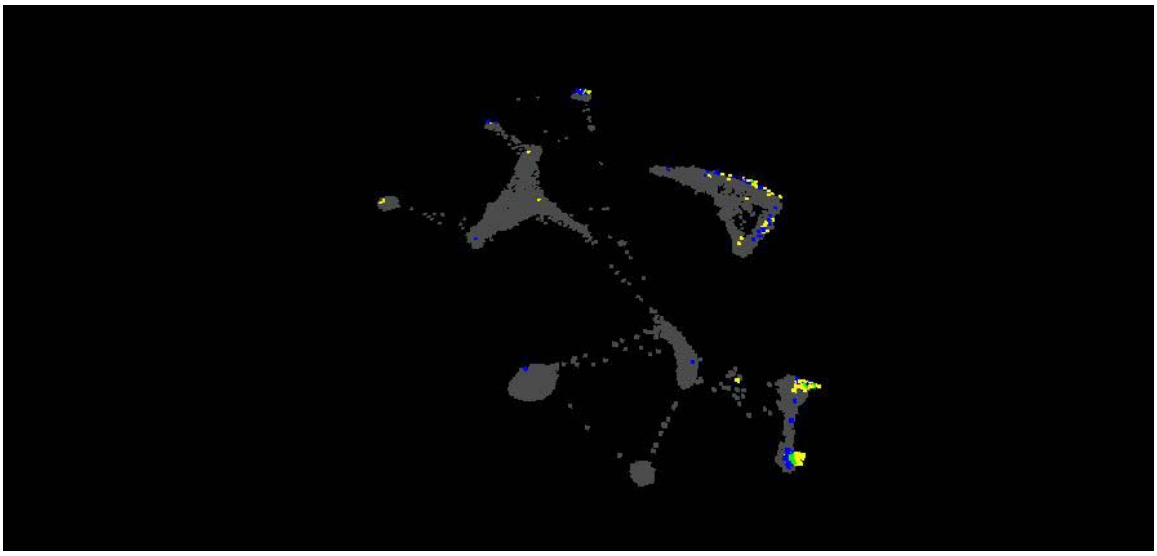


Figure 80. The general collocation of genes identified by different algorithms (shown with different colors). This collocations suggests that the different methods are in reasonable agreement.

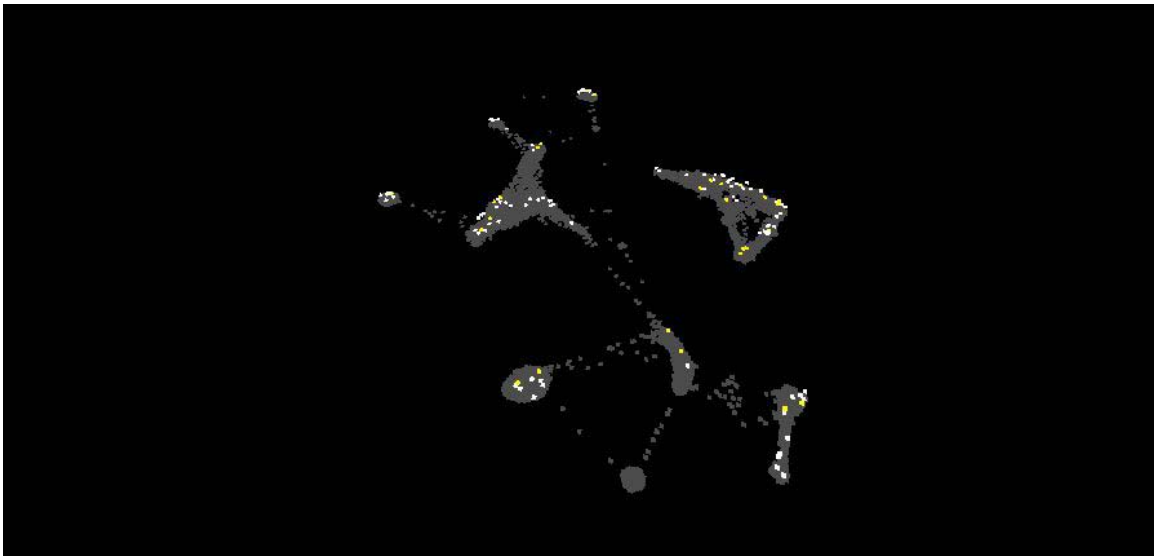


Figure 81. Here the genes selected by each method are widely separated and show no coherence, suggesting that there a lack of consensus among the methods.

When two methods produce similar gene lists the coherence may be due to the underlying similarity of the methods more than to any true biological significance, for instance Fisher's discriminant and the ANOVA methods are much more similar to each other than to Bayesian networks. Further, many methods will be heavily influenced by differences in the first few principle components of the gene expression data. On the other hand, methods, such as recursive elimination[49], perhaps aided by "boosting" [50] are able to examine the simultaneous efficacy of groups of genes, some of which, individually, may not be discriminatory in the first or second principle component. One way to understand these differences is by considering where selected genes project onto the plane of the first two principal components; see Figure 82, which schematically represents a few genes from three methods, identified by different colors.

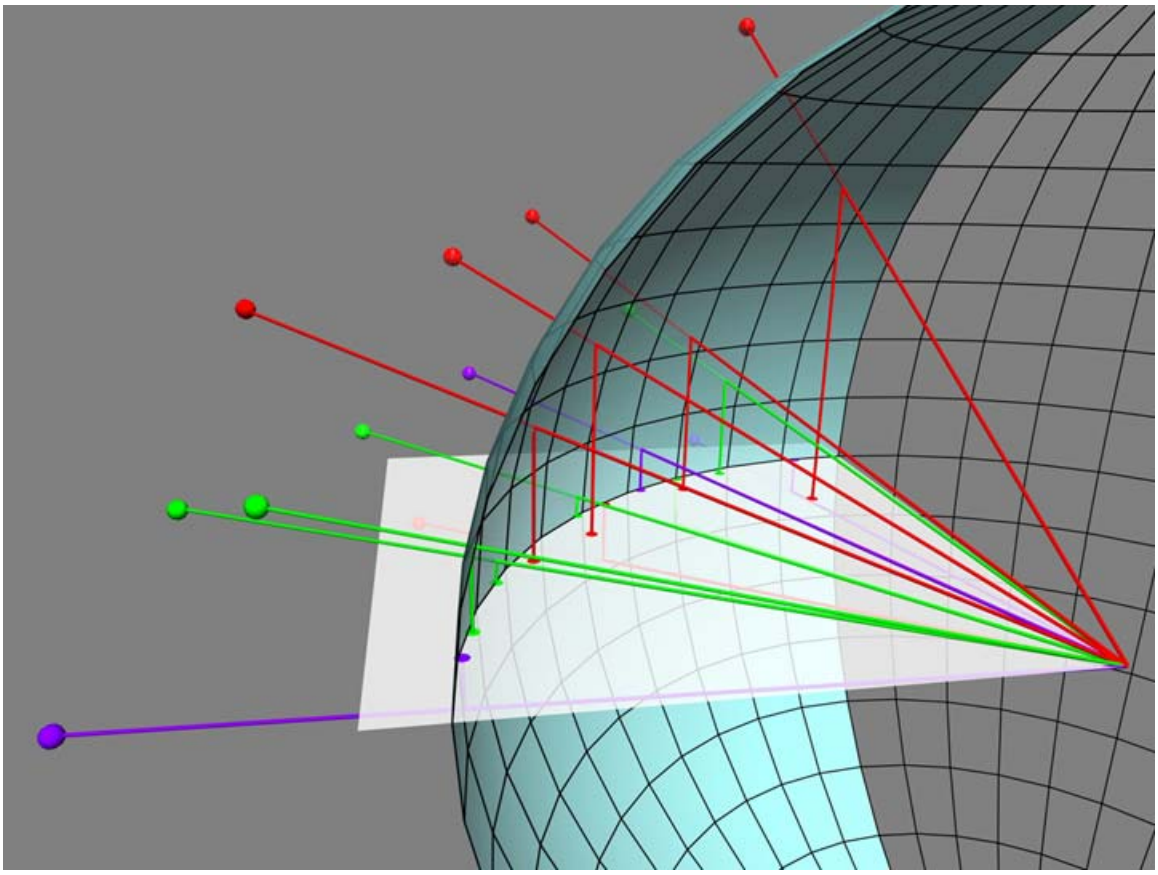


Figure 82. A few genes from three different methods are schematically shown intersecting the unit sphere and the projection of those intersections down onto the plane of the first two principle components. Note that genes near that plane will have projections that fall close to the arc of the sphere, while those above or below the plane with have intersections that fall well within the arc of the sphere.

In this approach, each gene is considered to be a point in patient-space, where each dimension corresponds to a different patient. Since, in this case, there were 12,625 genes and 126 patients, the spatial representation had 12,625 points (samples) in a 126 dimensional space. Of the 12,625 genes we only considered about 600 that occurred in the different gene lists, reducing our

problem to 600 genes in 126 dimensions. Furthermore, because we were mainly interested in how the genes compared as discriminators, and not how their actual expression levels compared, we projected the genes onto the 126 dimensional unit sphere in patient-space, as suggested in Figure 82. Geometrically, this corresponds to comparing the "directions" of the genes in the various gene lists as opposed to their "magnitudes".

In order to understand this visualization is it useful to imagine a sphere with a plane passing through the origin. The sphere corresponds to the unit sphere (the sphere with radius one centered at the origin) in the patient space and the plane corresponds to the plane determined by the first two principal components. The first principal component points in the radial direction of the sphere and the second principal component is tangential to the sphere at the sphere's intersection with the first principal component. It is precisely the first two dimensions that are shown in Figure 83. The vector representing a particular gene will intersect the unit sphere, and will be near the arc of the sphere (unit circle) in the plane if it lies in the first two principal components. To the extent that the gene lies above or below the plane of the first two principle components, the projection of the intersection back down onto the plane will lie further inside the arc. The distribution of these projections onto that principal component plane suggests how a given method of gene selection identifies important genes.

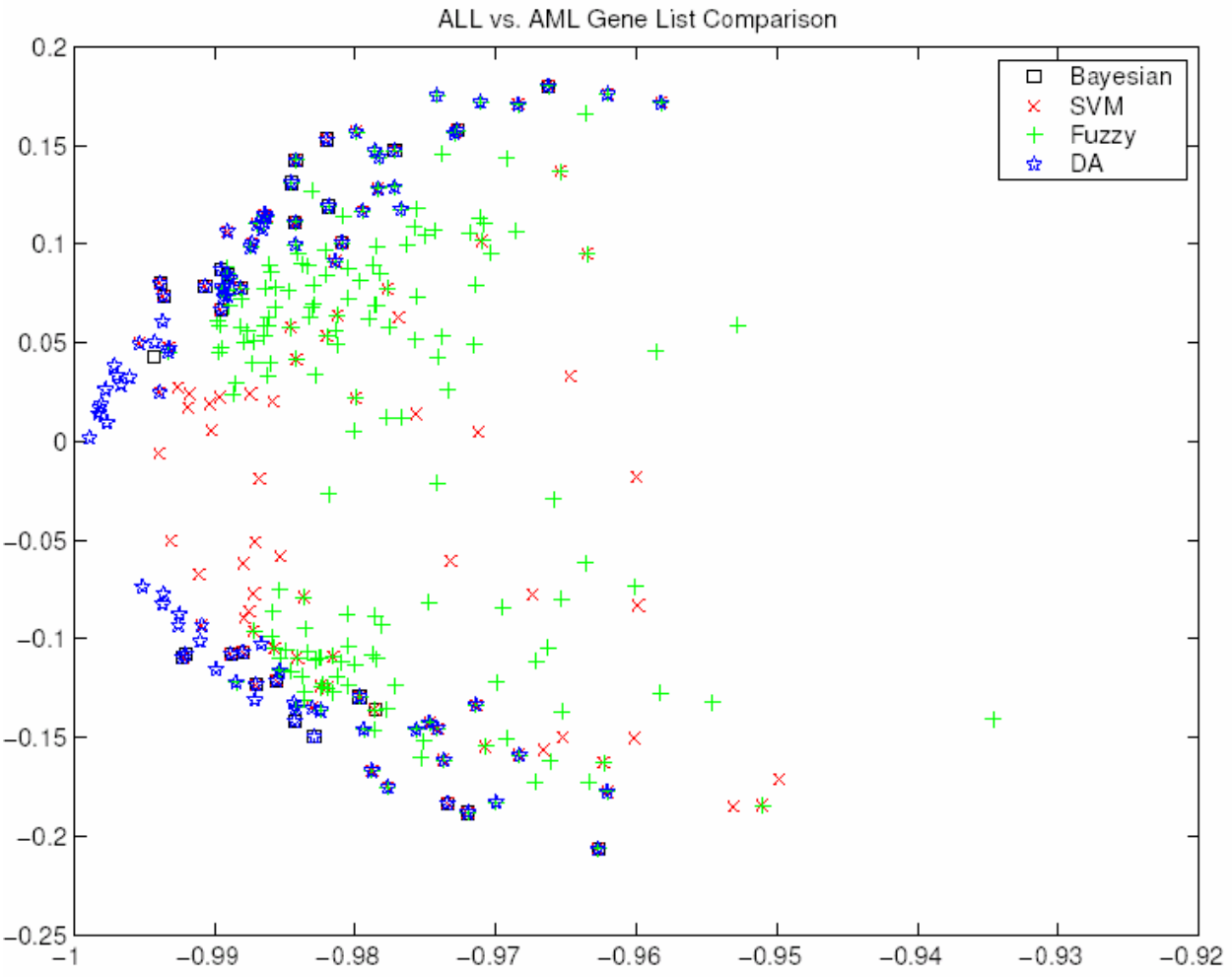


Figure 83. ALL vs. AML gene lists comparison. The gene lists that characterize ALL versus AML are shown, with a different color for each of the methods used to obtain them. In distinguishing infant ALL from infant AML we found that most of the genes in the list were co-localized in our representative visualization. Compare this plot with the results shown in Figure 84.

One of the main observations that can be made is the division of the gene lists above and below the center of the plot, Y=0. This division is especially noticeable in the Bayesian and discriminant analysis gene lists and is due to the fact that these methods are univariate gene selection methods. The univariate methods rank and subsequently select genes as isolated variables, and hence obtain gene lists that are in some sense very redundant. In contrast, the NeuroFuzzy and SVM methods are multivariate and tend to select gene lists that are less redundant and hence not entirely determined by the first two principal components.

It is evident from Figure 83 that the gene lists selected for the ALL/AML problem are related. Unfortunately, it is equally obvious that the gene lists selected for the remission/failure problem are unrelated, as shown using the same analysis in Figure 84.

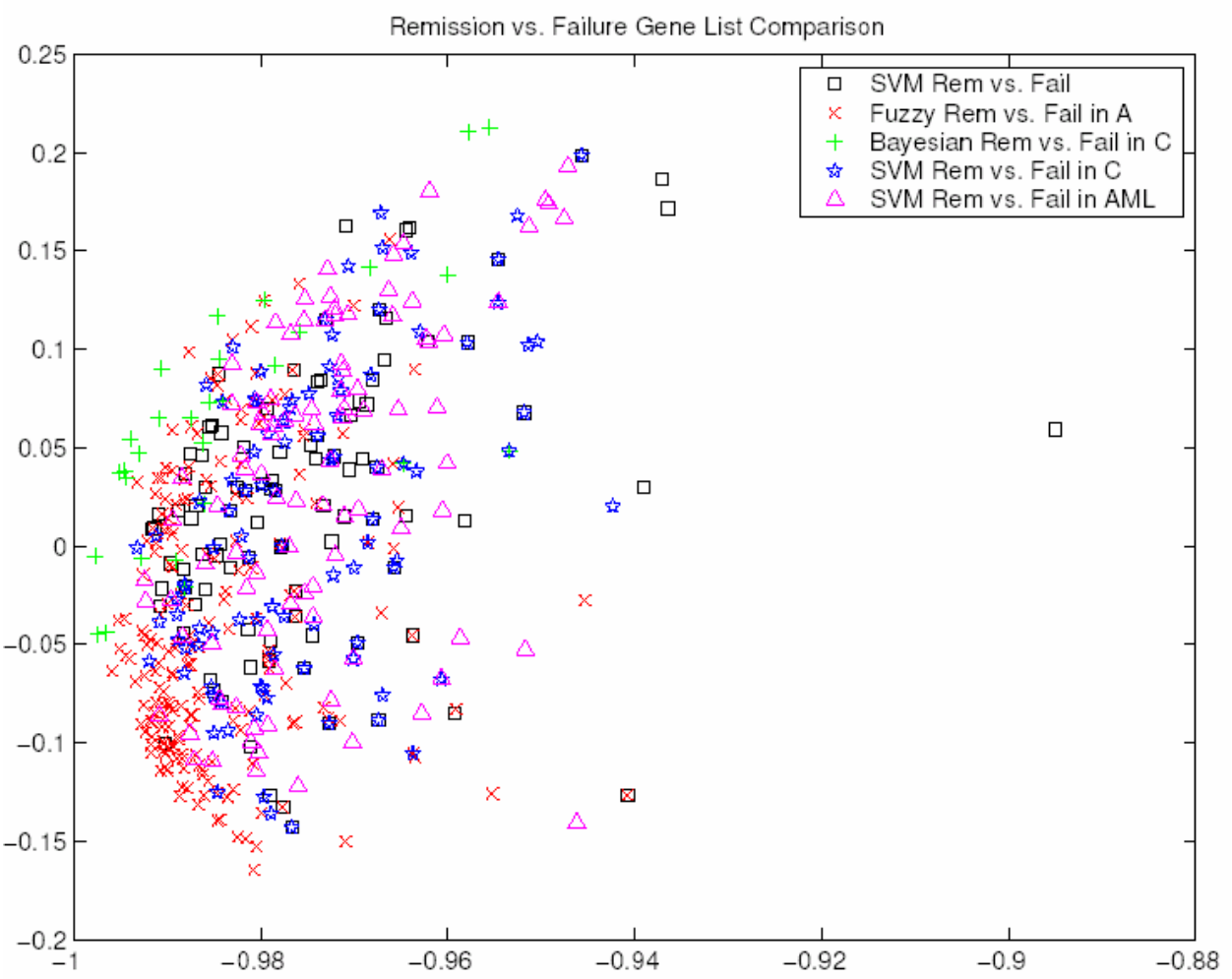


Figure 84. Remission vs. Failure gene lists comparison. The gene lists that characterize remission versus failure are shown, with a different color for each of the methods used to obtain them. It can be seen in this figure that distinguishing remission from failure is a difficult task.

When distinguishing infant ALL from infant AML we found that most of the list were co-localized in our representative visualization (see Figures 80 and 83). When distinguishing remission from failure, on the other hand, we could not arrive at a satisfactory conclusion (see, Figures 81 and 84), which is also consistent with the way the gene lists show up in Figures 80, and 81, and is also consistent with the results from ANOVA plus bootstrapping (data not shown).

At this point in the analysis it may seem that the biology has dissolved into a sea of numbers and statistical methods. However, these methods are our only guideposts when we begin reading the known information about the indicated genes. Without them we could easily waste very valuable time and people in the study of genes which are only weakly, if at all, related to the central questions of the research. Guided by these methods, we can approach the literature with greater confidence and are much more likely to see the important biology re-emerge in the gene annotations and the cited literature.

However, even after these statistical filters, this literature is vast, and is not organized to make our searching particularly easy. We have come to recognize that this step (where very knowledgeable scientists must read, and read, and read even further) is the critical, rate limiting one for our research. As a result, we have begun a fruitful collaboration with the Natural Language Processing (NLP) community to build tools that find, summarize, and reorder important parts of the available online literature to make that reading process simpler, and more focused toward our research needs. In the following section we will present and discuss our preliminary automatic Genome Literature Exploration Environment.

The gene list exploration environment (GLEE)

A particularly important next step in the traditional exploratory analysis of microarray data is the literature review and study to learn everything that is known about these genes, especially with respect to disease and biological pathways. We collaborated with computational linguists to build a knowledge-mining tool, which we regularly use in our analysis. This first implementation of our Gene List Exploration Environment (GLEE program) has been able to speed up our searches through the text describing the genes identified by any of our approaches. A demonstration version of GLEE, together with user documentation, is available from the Computing Research Laboratory web site:

http://aiaia.nmsu.edu/.

The input to the system is a list of gene identifiers from Affymetrix translated by the program to the equivalent OMIM gene identifier (See Figure 85, and further details at the OMIM web site:

http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=OMIM.

As shown in Figure 86, the relevant OMIM text is retrieved and re-ordered to match the criteria that we use for evaluating genes. This automated retrieval and reordering also employs text summarization. We are presently in the process of extending GLEE to use a subset of the NCI Enterprise Vocabulary Services, EV, which is a first step toward a more knowledge-based tool that will be implemented with semantic networks. Because so much of our knowledge about the functions, localizations, and clinical impacts of genes is encoded in published literature, and because the effort to incorporate that knowledge is so labor and knowledge intensive, we believe the application of NLP to our specific needs is a critical, and a still largely missing tool for genomic and proteomic investigations.

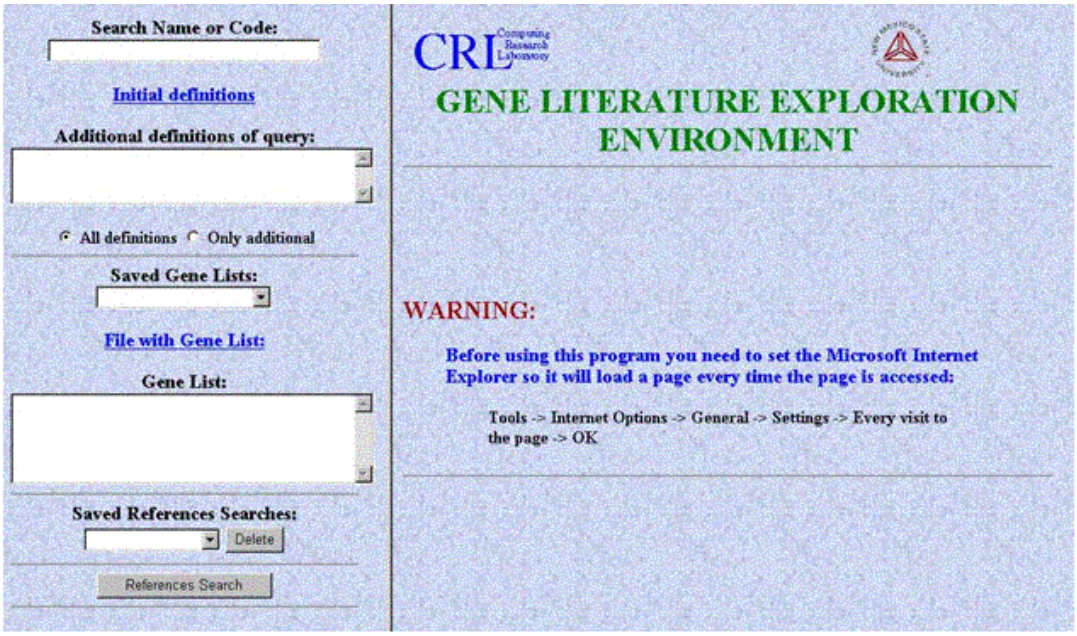


Figure 85. The Gene Literature Exploration Environment (GLEE) interface is configured as a web server, which handles document and query management, and a web browser that provides the user interface.

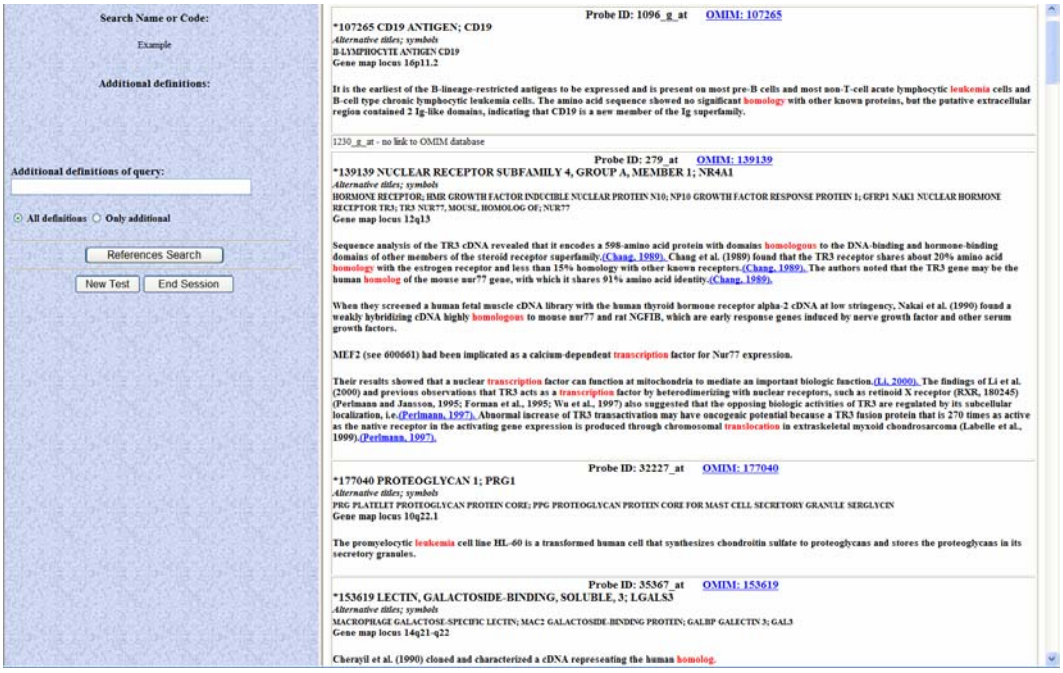


Figure 86. Output of the GLEE program. Summarized, and reordered annotations of a set of genes. Note that this is just the first page of annotations; further annotations are available by scrolling down in the browser.

Concluding remarks about the informatics methods

Exciting preliminary gene expression profiling studies are providing new insights into molecular mechanism, and hold the promise of deeper biological understanding. However, the speed at which groups of genes generated by microarray analysis can be put together in pathways is one of the limiting steps in the translation of these discoveries to applications.

The methods presented here can potentially be useful in uncovering groups of genes that serve to fingerprint biologically important subtypes that could aid further biological discoveries and in refining diagnosis and improving assessment of prognosis. Additionally, gene list comparison and exploration methods will increase the speed at which researchers can visualize and extract the more complex relationships encoded in gene expression data.

References

1. Schena, M., *DNA Microarrays: A Practical Approach*, in *The Practical Approach Series*, B.D. Hames, Editor. 1999, Oxford University Press, Inc.: New York. p. 210.
2. Brown, P.O. and D. Botstein, eds. *Exploring the new world of the genome with DNA microarrays*. Nature Genetics. Vol. 21(SS). 1999. 33-37.
3. DeRisi, J., et al., *Use of a cDNA microarray to analyse gene expression patterns in human cancer*, in *Nature Genetics*. 1998. p. 457-460.
4. van't Veer, L.J., et al., *Gene expression profiling predicts clinical outcome of breast cancer*, in *Nature*. 2002. p. 530-536.
5. Lee, M.-L.T., et al., *Importance of replication in microarray gene expression studies: statistical methods and evidence from repetitive cDNA hybridizations*, in *Proceedings of the National Academy of Sciences*. 2000. p. 9834-9839.
6. Schena, M., *Microarray Biochip Technology*. 2000, Eaton Publishing: Natick. p. 298.
7. Martinez, M.J., et al., *Identification and removal of contaminating fluorescence from commercial and in-house printed DNA microarrays*, in *Nucleic Acids Research*. 2003. p. in press.
8. Schultz, R.A., et al., *Hyperspectral imaging: A novel approach for microscopic analysis*, in *Cytometry*. 2001. p. 239-247.
9. Bogdanov, V., *In-line complete hyperspectral fluorescent imaging of nucleic acid molecules*. 2001, Orchid BioSciences, Inc.: USA.
10. Garner, H.R., *Hyperspectral slide reader*. 2000, Board of Regents, The University of Texas System: USA.
11. Bro, R. and S. DeJong, *A fast non-negativity-constrained least squares algorithm*. Journal of Chemometrics. Vol. 11. 1997. 393-401.
12. Tauler, R., E. Casassas, and A. Izquierdoridorsa, *Self-Modeling Curve Resolution in Studies of Spectrometric Titrations of Multi-Equilibria Systems by Factor-Analysis*, in *Analytica Chimica Acta*. 1991. p. 447-458.
13. Tauler, R., A. Izquierdoridorsa, and E. Casassas, *Simultaneous Analysis of Several Spectroscopic Titrations with Self-Modeling Curve Resolution*, in *Chemometrics and Intelligent Laboratory Systems*. 1993. p. 293-300.
14. Tauler, R., et al., *Application of a New Multivariate Curve Resolution Procedure to the Simultaneous Analysis of Several Spectroscopic Titrations of the Copper(Ii)-Polyinosinic Acid System*, in *Chemometrics and Intelligent Laboratory Systems*. 1995. p. 163-174.
15. Tauler, R., *Calculation of maximum and minimum band boundaries of feasible solutions for species profiles obtained by multivariate curve resolution*, in *Journal of Chemometrics*. 2001. p. 627-646.
16. Sinclair, M.B., et al., *Design, construction, characterization, and application of a hyperspectral microarray scanner*, in *Applied Optics*. 2003. p. submitted.
17. Haaland, D.M., R.G. Easterling, and D.A. Vopicka, *Multivariate Least-Squares Methods Applied to the Quantitative Spectral-Analysis of Multicomponent Samples*, in *Applied Spectroscopy*. 1985. p. 73-83.
18. Haaland, D.M. *Multivariate calibration methods applied to quantitative FT-IR analyses*. in *Practical Fourier Transform Infrared Spectroscopy*. 1990. San Diego: Academic Press.

19. Jolliffe, I.T., *Pincipal Component Analysis*. 1986, Springer-Verlag: New York.
20. Van Benthem, M.H., M.R. Keenan, and D.M. Haaland, *Application of equality constraints on variable during alternating least quares procedures*, in *Journal of Chemometrics*. 2002. p. 613-622.
21. Timlin, J.A., et al., *Hyperspectral microarray scanning: Impact on accuracy and reliability of genomic data*, in *Nature Biotechnology*. in preparation.
22. Brown, C.S., P.C. Goodwin, and P.K. Sorger, *Image metrics in the statistical analysis of DNA microarray data*, in *Proceedings of the National Academy of Sciences of the United States of America*. 2001. p. 8944-8949.
23. Kegelmeyer, L.M., et al., *A groundtruth approach to accurate quantitation of fluorescence microarrays*, in *Microarrays: Optical Technologies and Informatics*, E.R. Dougherty, Editor. 2001, SPIE - The International Society for Optical Engineering: San Jose, CA. p. 35-45.
24. Thompson, J.R., *Simulation: a modeler's approach.* 2000: p. xv.
25. Davidson, G., et al., *Knowledge mining with VxInsight: Discovery through interaction.* JOURNAL OF INTELLIGENT INFORMATION SYSTEMS, 1998. **11**(3): p. 259-285.
26. Davidson, G., B. Wylie, and K. Boyack. *Cluster stability and the use of noise in interpretation of clustering*. in *7th IEEE Symposium on Information Visualization (INFOVIS 2001)*. 2001. SAN DIEGO, CALIFORNIA.
27. Tukey, F.M.a.J.W., *Data analysis and regression*. Addison-Wesley Series in Behavioral Science: Quantitative Methods, ed. F. Mosteller. 1977, Reading Massachusetts: Addison-Wesley Publishing Company.
28. Savage, I.R., *Contributions to the theory of rank order statistics-the two-sample case.* Annals of Mathematical Statistics, 1956. **27**: p. 590-615.
29. Wilcox, R.R., *Fundamentals of modern statistical methods: substantially improving power and accuracy*. 2001, New York, NY: Springer-Verlag, Inc. p. 110.
30. Wilcox, R.R., *Fundamentals of modern statistical methods: substantially improving power and accuracy*. 2001, New York, NY: Springer-Verlag, Inc. p. 113.
31. Wilcox, R.R., *Introduction to robust estimation and hypothesis testing*. Statistical Modeling and Decision Science, ed. G.J.L.a.I. Olkin. 1977, San Diego, CA.: Academic Press. p. 188.
32. *Online Mendelian Inheritance in Man, OMIM(tm)*. 2000, McKusick-Nathans Institute for Genetic Medicine; National Center for Biotechnology Information, National Libary of Medicine.
33. Kim, S., et al., *A gene expression map for Caenorhabditis elegans.* SCIENCE, 2001. **293**(5537): p. 2087-2092.
34. Fisher, R.A., *On the probable error of a coefficient of correlation deduced from a small sample.* Metron, 1921. **1**(4): p. 3.
35. Ostel, B., *Statistics in research: basic concepts and techniques for research workers*. Second ed. 1963, Ames, Iowa: The Iowa State University Press. 226.
36. Werner-Washburne, M., et al., *Comparative analysis of multiple genome-scale data sets.* GENOME RESEARCH, 2002. **12**(10): p. 1564-1573.
37. Eades, P., *A heuristic for graph drawing. Congressus Numerantium*, 1984. **42**: p. 149-160.
38. T. Fruchtermann and E. Rheingold, *Graph drawing by force-directed placement*. 1990, University of Illinois: Urbana-Champagne, Il.

39. Quinn, N. and M. Breur, *A force directed component placement procedure for printed circuit boards.* IEEE Transactions on Circuits and Systems, 1979(6): p. 377-388.
40. Otten, R. and L. van Ginneken, *The annealing algorithm*. 1989, Boston, MA: Kluwer Academic Publishers.
41. Kamada, T. and S.Kawai, *Automatic display of network structures for human understanding*. 1988, Tokyo University: Tokyo Japan.
42. Davidson, R. and S. Kawai, *Drawing graphs nicely using simulated annealing*. 1989, The Weizmann Institutue: Rehovot, Israel.
43. Kamada, T. and S. Kawai, *An algorithm for drawing general undirected graphs.* Information Processing Letters, 1989. **1**(31): p. 7-15.
44. Kamada, T., *A simple method for computing general position in displaying three-dimensional objects.* Computer Vision, Graphics, and Image Processing, Kawai, S. **41**: p. 43-56.
45. KIRKPATRICK, S., C. GELATT, and M. VECCHI, *OPTIMIZATION BY SIMULATED ANNEALING.* SCIENCE, 1983. **220**(4598): p. 671-680.
46. Spellman, P., et al., *Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization.* MOLECULAR BIOLOGY OF THE CELL, 1998. **9**(12): p. 3273-3297.
47. Eisen, M., et al., *Cluster analysis and display of genome-wide expression patterns.* PROCEEDINGS OF THE NATIONAL ACADEMY OF SCIENCES OF THE UNITED STATES OF AMERICA, 1998. **95**(25): p. 14863-14868.
48. EFRON, B., *COMPUTER-INTENSIVE STATISTICAL-INFERENCE - A CITATION CLASSIC COMMENTARY ON BOOTSTRAP METHODS - ANOTHER LOOK AT THE JACKKNIFE BY EFRON,B.* CURRENT CONTENTS/PHYSICAL CHEMICAL & EARTH SCIENCES, 1989(37): p. 16.
49. I.Guyon, et al., *Gene selection for cancer classification using support vector machines.* Machine Learning, 2002. **46**: p. 389-422.
50. Freund, Y. and R.E. Shapire, *A short introduction to boosting.* Journal for Japanese Society for Artificial Intelligence, 1999. **14**(5): p. 771-780.

Distribution:

(pdf version available from the SNL Library web page)

5	MS-0318	George Davidson, 9212
1	MS-0323	LDRD Offie, 1011
3	MS-0886	David M. Haaland, l812
1	MS-9018	Central Technical Files, 8945-1
2	MS-0899	Technical Library, 9616