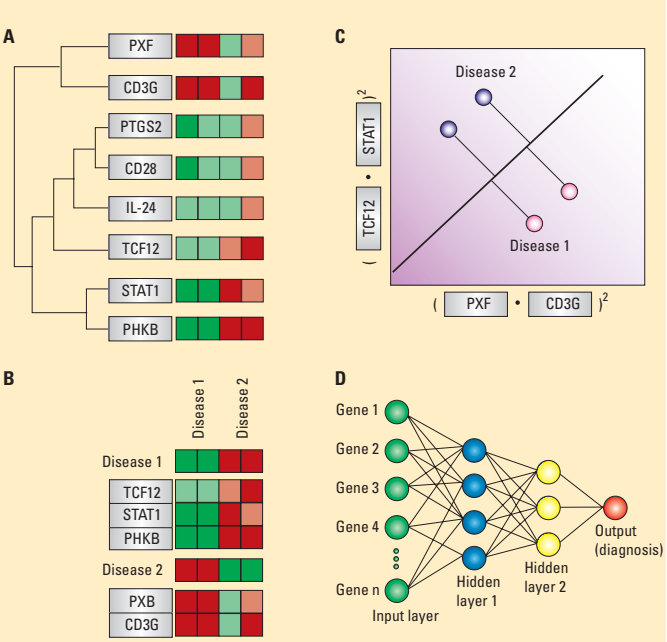
Gene expression learned

"Supervised" statistical techniques are gaining popularity for microarray analysis.

BY DAVID FILMORE

Software, both freeware and commercial, is continually evolving to tackle the mass of microarray expression data to profile disease or drug activity, or, conversely, to classify genes based on shared functions or regulations.

It has only been in the past 5 or 6 years that sophisticated multivariate statistical techniques have been brought to bear on analyzing microarray data, which now has progressed to the possibility of genomewide analysis on a single chip. In that time, some techniques, most notably hierarchical clustering and self-organized maps (SOMs), have become standards in the field and are components of just about every software package used for gene expression analysis. These methods are generally considered to be unsupervised techniques, meaning they work with purely statistical considerations of variation and similarity to find internal structure or relationships in the expression patterns. The results can subsequently be looked at in a biological context that could lead to conclusions with mechanistic, diagnostic, or therapeutic significance.



Distinguished data. Representations of some unsupervised and supervised microarray data analysis techniques. (A) Hierarchical clustering, (B) nearest-neighbor analysis using hypothetical expression patterns, (C) support vector machine using mathematical combinations of genes to construct a high-dimensional hyperplane, and (D) a neural network. (Parts A, B, and C are adapted with permission from Ref. 6.)

Hierarchical clustering works by iteratively grouping together genes (or samples) that are highly correlated in terms of their expression levels, then correlating the groups themselves to arrive at a diagram (specifically, a treelike dendrogram) in which the genes are lined up such that similar ones appear next to each other. Michael Eisen, Patrick Brown, and others at the Stanford University School of Medicine originated this approach to microarray analysis and software (*http://rana.lbl.gov/EisenSoftware.htm*) and reported on it in a 1998 paper (*1*).

SOMs are essentially plots of expression data in which each microarray-analyzed sample is an axis, gene expression levels act as the coordinates, and distance between points indicates some measure of statistical similarity. The result is that genes that are similarly expressed in the different samples cluster together, and patterns on the multidimensional space might illustrate important distinctions between samples from different sources. Researchers at the Whitehead Institute, Massachusetts Institute of Technology (MIT), and Harvard University, including Todd Golub and Eric Lander, established the effectiveness of this approach with their GeneCluster software (*www.broad.mit.edu/cancer/software/software.html*) in 1999 (*2*).

These approaches have great promise for providing more precise discriminations in the diagnostic, prognostic, and therapeutic response categories than would otherwise be possible. But as DNA arrays have become a more entrenched component of biomedical research and large gene expression profiles more readily attainable, there has been movement toward including supervised methods of analysis, in which preliminary knowledge is intrinsically factored into the analysis. In this way, microarray results can be used to make direct biological-based class predictions instead of illustrating purely statistical relationships. Techniques that rely on training or learning processes that computationally match input combinations with certain outputs have the potential to answer a specific question, such as, does this patient have condition X, and is she expected to respond to treatment X, based on her genomic response?

"Near" predictions

One of the earliest and most highly cited examples of supervised class prediction was the distinction made between acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) by the Golub–Lander team (*3*). The researchers, using what is called a "nearest-neighbor" approach, compared the gene expression profiles (of 6817 genes) from 38 leukemia samples to an "idealized" pattern that includes genes that show uniformly high expression in one class and uniformly low expression in the other. They found many genes that were highly correlated with this

model expression pattern. For the 50 best-correlated of these genes, they determined the threshold level of expression for optimal class prediction with training samples. These parameters were then used to accurately predict the AML/ALL classification of independent (i.e., test) samples.

A similar approach is called nearest-centroid classification, in which a training set is used to compute, for each class, a standardized "centroid" (i.e., the average expression for each gene in each class divided by the within-class standard deviation for that gene). Gene expression profiles of new samples can then be compared with the centroids, and the one that they are closest to is their predicted class. Robert Tibshirani (*www-stat.stanford.edu/~tibs*) and colleagues at the Stanford statistics department developed a refined version of this method they called nearest-*shrunken* centroid classification, in which each centroid is reduced in size by a certain threshold, thus pinpointing gene expression changes that play the strongest role in class differentiation and using only these genes for prediction. The scientists recently demonstrated this strategy, which they have packaged as a software tool called Prediction Analysis of Microarrays (PAM), for classifying small round blue cell tumors and leukemias (*4*). It is based on previous, and widely used, software they had developed called Significance Analysis for Microarrays (SAM), which applies a set of gene-specific t-tests to identify statistically significant changes in expression.

Techniques such as nearest neighbor and PAM are highly attractive for their simplicity. But awareness of the complex workings of the genome has triggered interest in applying even more computationally sophisticated techniques to the task of unearthing useful knowledge from microarray experiments, namely machine learning tools that have the potential to pick up on more subtle or convoluted correlations in expression profiles. Three such algorithms that have received particular attention are classification trees, support vector machines (SVMs), and neural networks.

Machine learning

Classification trees are formed by the process of recursive partitioning, in which attributes of a data set are judged by their ability to distinguish between classes of information. Training samples can be used to construct a top-down flowchart, or tree, consisting of a series of questions or tests that start with all observations (e.g., the total expression profile from a microarray) and, hopefully, end with separate categories for each class.

Erik Gunther and colleagues at the pharmaceutical company CuraGen recently demonstrated the power of this strategy as a tool to predict clinical drug efficacy from genomic expression profiles (*5*). They classified known drugs or candidates as antidepressants, antipsychotics, or opioid receptor agonists purely on the basis of microarray results from neuron cells exposed to drugs in these categories. They performed a classification tree algorithm and achieved 89% correct classification.

Awareness of the complex workings of the genome has triggered interest in applying even more computationally sophisticated techniques.

Notably, even if a particular subset of one of the drug classes was not included in the training samples, it was still accurately classified in the testing of the model. For example, when no SSRI (selective serotonin reuptake inhibitor) antidepressants were included in the training set, the tree still classified such a drug as an antidepressant.

SVMs offer the advantage of expanding the number of features available for differentiating sets of data by combining genes into mathematical functions (called kernel functions). According to a review article by Atul Butte from the informatics program at Children's Hospital in Boston, "It is possible that even if genes A and B individually could not be used to separate the two sets of biological samples, together with the proper kernel function, they might successfully separate the two" (*6*). Training samples for SVMs are used to define a hyperplane that best separates two classes in the extremely high dimensional feature space constructed from the kernel functions. The further to one side of the plane a test sample is, the higher the confidence in the prediction.

The Golub–Lander team demonstrated the usefulness of this approach by utilizing the MIT-developed SVM-FU software (*http://five-percent-nation.mit.edu/SvmFu*) for multiclass cancer diagnosis. Their attempts to categorize samples from patients with 14 different types of cancers by unsupervised analysis (hierarchical clustering and SOM) of expression profiles from 16,063-gene microarrays led to substantial intermixing of the different cancers (*7*). But by generating an SVM algorithm with 14 different hyperplane classifiers (each test sample was presented sequentially to the cancer-specific classifiers to answer questions such as "Breast cancer or not breast cancer?"), the scientists correctly predicted the diagnosis of 78% of 54 test samples, compared with 65% from nearest-neighbor analysis and 45% from a weighted-voting approach. It was observed that the expressions of a large portion of the 16,063 genes played some role in the multifaceted distinction process, although the genes most highly correlated with each of the 14 tumor classes were identified.

Neural networks. Loosely modeled on the parallel functioning of the mammalian brain, neural networks have gained a lot of attention for their ability to capture highly nonlinear relationships within data. They consist of a multilayer system, including input neurons, output neurons, and several "hidden" neuron layers that make up the convoluted learning path in between. Statisticians tend to be wary of neural networks because, to perform the nonlinear analysis, they need to put to use so many parameters that overfitting the data (essentially, memorizing the training set, which precludes accurate generalization of new samples) becomes a concern (*8*). But for massive amounts of data with poorly understood interrelationships, the pluses may outweigh the negatives.

Scientists at the University of Maryland School of Medicine, led by Yan Xu, recently used a neural network to distinguish between the gene expression profiles of

esophageal cancer and its premalignant condition, called Barrett's esophagus (*9*). After a gene-filtering process using the SAM program, they constructed the network using 12 training samples and achieved 100% classification accuracy for 10 test samples. Because esophageal cancer is usually discovered at an advanced stage and is rapidly fatal, this distinction is important for diagnostics.

Supervised software

As supervised methods of analysis get more proof-of-principle attention, as in the examples discussed here, they are being incorporated into the software packages widely used for new genomic research endeavors. MIT's GeneCluster has nearest-neighbor algorithms included in its most recent version with the unsupervised clustering analysis components, as does Silicon Genetics' (*www.silicongenetics.com*) GeneSpring software. Biodiscovery's (*www.biodiscovery.com*) GeneSight also includes neural network clustering in its package. In addition, general (not bioinformatics-specific) statistics packages, such as the freely downloadable "R" packages (*www.r-project.org*) and commercial software such as S-plus, Matlab, and MiniTab, are being applied for a range of unsupervised and supervised methods in microarray studies.

Obviously, the method (or methods) most appropriate for a particular problem will vary, and thus the availability of a wide assortment of analysis tools, as well as the ability to perform other functions such as storing and normalizing the data, is important for a broadly effective software suite.

References

(1) Eisen, M. B.; et al. *Proc. Natl. Acad. Sci. U.S.A.* **1998,** *95,* 14863–14868.
(2) Tamayo, P.; et al. *Proc. Natl. Acad. Sci. U.S.A.* **1999,** *96,* 2907–2912.
(3) Golub, T. R.; et al. *Science* **1999,** *286,* 531–537.
(4) Tibshirani, R.; et al. *Proc. Natl. Acad. Sci. U.S.A.* **2002,** *99,* 6567–6572
(5) Gunther, E. C.; et al. *Proc. Natl. Acad. Sci. U.S.A.* **2003,** *100,* 9608–9613
(6) Butte, A. *Nat. Rev. Drug Discov.* **2002,** *1,* 951–960.
(7) Ramaswamy, S. *Proc. Natl. Acad. Sci. U.S.A.* **2001,** *98,* 15149–15154.
(8) Amaratunga, D.; Cabrera, J. *Exploration and Analysis of DNA Microarray and Protein Array Data;* Wiley-Interscience: New Jersey, 2004.
(9) Xu, Y.; et al. *Cancer Res.* **2002,** *62,* 3493–3497. ■