Discovering Patterns in Microarray Data

HARRY B. BURKE, MD, PhD

Valhalla, New York

The human genome is a complex system characterized by gene interactions and nonlinear behaviors. Complex systems cannot be viewed as the aggregate of their isolated pieces but must be studied as an integrated whole. Microarray technologies offer the opportunity to see the entire biological system as it existed at one moment in time. It is tempting to try to analyze the entire microarray at once to immediately discover the pattern being sought, for example, the pattern of a breast cancer. However, such an analysis would be a mistake because microarrays provide massively parallel information, the analysis of which is a nondeterministic polynomial time (NP)-hard problem. Current statistical methods are not sufficiently powerful to solve this NP-hard problem. The best approach to microarray analysis is to begin with a small number of the elements in the microarray known to be a pattern and ask questions of the other elements in the microarray; i.e., perform instantaneous scientific experiments regarding whether each of the other elements in the microarray are related to the known pattern.

Key words: predictive medicine, microarray, principal components analysis, clustering, self-organizing maps, artificial neural networks, pattern discovery, pattern recognition, nondeterministic polynomial time-hard problem.

The ultimate goal of disease-related proteogenomic (the term "proteogenomic" was coined by Cooper [1]) research is a complete description of the unfolding of a specific disease process from its time of origin (Fig. 1). At a particular moment in the disease's biological development, this description includes a representation of all the necessary and sufficient genes (the genome), systematic linkage of the genomic representation to a representation of the transcription of these genes (transcriptome), and systematic linkage of the transcriptome to a representation of the resulting proteins (proteome). For a complete understanding of the disease process, each description of a time in the disease process must then be linked to the other descriptions at successive moments in time.

As this model is clinically realized, it will become possible to target interventions at specific times and etiologic locations in the disease process. It will also be possible to accurately predict both the direction and magnitude of the changes that will occur in response to the intervention. For example, it will allow one to know the patient's stage of the disease process and allow the creation of a therapy that acts at a particular time and place in the disease process.

When properly analyzed, large-scale microarrays have the potential to provide the data necessary to determine the disease-specific and patient-specific relationships of genome to transcriptome to proteome. Specifically, microarrays will be used to explore the genetic mechanisms that give rise to a disease. In addition, they will be used, without knowledge of their role in the disease process, as

From the Bioinformatics and Computational Research Group, New York Medical College, Valhalla, NY.

Reprint requests: Harry B. Burke, MD, PhD, New York Medical College, Department of Medicine, Valhalla, NY 10595. Email: harry_burke@nymc.edu

1084-8592/00/0504-0011$10.00/0

doi:10.1054/modi.2000.19562

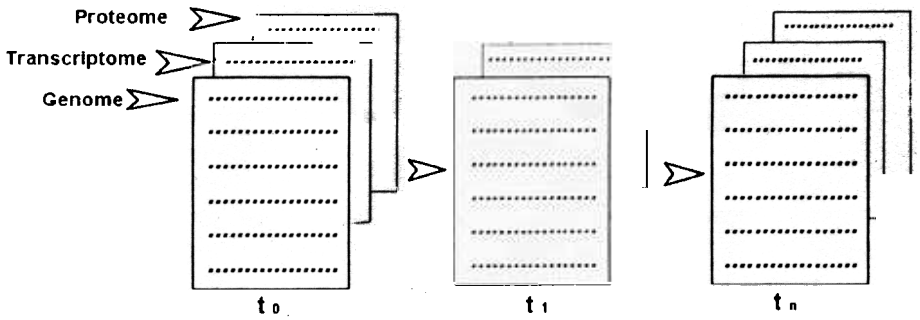

Fig. 1. Unfolding of the disease process. Each set of three microarrays represents the disease process at that point in time in terms of the changes in the genome, transcriptome, and proteome.

predictive factors. Depending on the type of predictive factor, it will allow the determination of (1) the risk for disease, (2) existence of disease, and (3) prognosis without and with treatment [2]. On a practical level, pharmacogenomics is the search for the genetic factors that predict the patient's response to a specific treatment, or the discovery of genetic patterns that are patient and therapy specific (a patient's profile for that therapy). These patterns will predict whether an individual patient with a particular disease will respond to a specific treatment and whether that treatment will cause toxic side effects in the patient [2].

This report addresses some issues that arise when analyzing information generated by microarray technologies. Although the genetics of cancer is the focus of this report, the concepts are applicable to most disease-related proteogenomic analyses.

Information Theoretic Perspective and Massively Parallel Information

A large-scale microarray containing as few as several hundred genes to as many as one hundred thousand genes creates an analog representation of the average activity level of a gene across a huge number of cells for a large number of genes at a biological moment in time for a particular patient. The output of a microarray contains signal, the true activity level of each gene, and noise, the spurious and background activity level.

Microarrays are a source of massively parallel information. They are able to generate large amounts of information in a nonserial manner; the information is not the result of a conditional sequence of investigative events. There are few examples of massively parallel information in science; thus, little is known about its analysis.

Magnetic resonance imaging might be relevant to microarrays because it generates large amounts of information in parallel in an image-processing paradigm. However, its task is made orders of magnitude simpler by image processing with a preexisting natural distance metric, i.e., spatial proximity. In the microarray domain, this is similar to genes being arranged *a priori* on the chip in the natural order of their decreasing relatedness to a particular biological function.

It is reasonable to ask why we should deal with the problems inherent in massively parallel information. Simply stated, massively parallel information is essential to the understanding of complex diseases. Because microarrays represent information in a massively parallel fashion, they can simultaneously present multiple components of a complex system. Before the availability and use of microarrays, molecular genetic research proceeded in a stepwise fashion from one biological component to the next. Although conditional analysis may be useful for simple systems, it is not a viable approach when dealing with complex (nonlinear, interactional), tightly integrated, dynamic systems. Such systems must be examined and understood as a whole rather than as a series of pieces.

Three characteristics of complex dynamic systems are: (1) they possess alternative pathways, i.e., there are different ways of accomplishing the same result; (2) each component has multiple functions, i.e., different activities that it can perform; and (3) pathways and functions are determined by the activities of other components of the system. This type of system cannot be successfully studied in pieces because when the study is completed, the pieces will not fit together; each piece has a differ-

ent context (environment). In other words, cancer is not like a jigsaw puzzle.

Pattern Discovery

We are not interested in discovering single genes; rather, we are interested in patterns because patterns have the greatest functional significance. Pattern can be operationally defined as a set of elements that occurs in a manner that is systematic and meaningful for the task. In the context of microarrays, there are two types of pattern tasks. Pattern discovery is the detection, or more correctly, the learning of a new pattern from the data. Pattern recognition is the recognition of a pattern when it recurs, i.e., the ability to identify a pattern as an instance of a known pattern. Pattern discovery and recognition are ancient problems; their literature extends back to the ancient Greek philosophers. Currently, they are a central problem in the psychology of human perception. Pattern recognition is the less challenging problem because after a pattern is known, templates and other approaches can be easily applied. The remainder of this report is devoted to pattern discovery.

Computationally Intractable Problems

Initially, it must be assumed that every data element in a massively parallel information representation has the potential to be meaningful, i.e., to be a necessary but not sufficient part of the pattern. The reason for this assumption is that if it were not possible for each data element to be meaningful, then massively parallel information would not be necessary. It is precisely because any element could be important that we are interested in and willing to deal with the problems of massively parallel information. Typically, microarray data contain thousands of elements per patient and few patients [3]. Microarrays present an analytic problem that is nondeterministic polynomial time (NP)-hard. NP represents a class of computational tasks for which a potential solution can be checked efficiently for correctness, yet finding such a solution appears to require exponential time in the worst case. In other words, they are currently computationally intractable.

Simplifying the Problem to Make It More Tractable

The analysis of large-scale microarray-generated information to find true patterns is intractable when (1) every gene is considered to be a continuous variable and there are (2) thousands of genes, (3) interpatient variation, (4) disease variation (stage, subtype), (5) error in the microarray technology, and (6) only a few exemplars of the pattern, i.e., only a few individuals express the pattern in the data.

The analysis of large-scale microarrays can be simplified by thresholding each gene's signal to create binary variables (which consequently have reduced information available), minimizing disease and interpatient variation, and increasing the number of patients. However, even in this situation, because of splice variants and other issues, there are more genes than one would like to analyze. In the simplest of conditions, in which the genes are considered binary variables, there are 2^n possible patterns (where n is the number of genes and their variants). However, 2^n is still a very large number. Each gene is considered a dimension in the analysis, and all the patients' values for that gene define the boundaries and shape of the dimension. This is very high-dimensional space that has its own characteristics, for example, the curse of multidimensionality. High-dimension space is extremely large, and each patient's data move to the edges of the dimensions.

A potentially useful heuristic for dimension reduction is differential expression, which uses a subtraction strategy [4–8]. This approach usually involves subtracting normal gene expression from the gene expression of cancer cells to identify relatively overexpressed and underexpressed genes. This heuristic should not be confused with pattern discovery algorithms.

Statistical Methods for Pattern Discovery

The difficulty inherent in analyzing microarrays has been discussed. We now explore the ability of the current statistical methods designed for relatively simple problems to deal with microarray data. To clarify our nomenclature, data mining used in the context of microarrays is almost always statis-

tical analysis. To date, almost all published microarray analyses have been based, either in part or completely, on traditional statistical methods. None has used a completely new method specifically designed for microarray analysis. Table 1 lists some of the published statistical analysis methods.

There are two categories of pattern discovery methods: unsupervised and supervised learning algorithms. In unsupervised learning, the final error metrics are not available during training; thus, the algorithm is not guided by an outcome. This approach has been called blind separation because there is no dependent variable. The task is to reduce the data complexity with minimal loss in precision by discarding noise and showing basic structures. The algorithms accomplish this by optimizing a cost function that preserves the original data as completely as possible while simultaneously favoring prototypes with minimal complexity. Unsupervised learning algorithms tend to focus on discovering linear relationships or maximizing signal to noise ratios and usually assume Gaussian distributions. Examples of unsupervised learning algorithms include principal components analysis (PCA), self-organizing maps (SOM), and some clustering algorithms. Most unsupervised learning algorithms can be used for dimension reduction, as well as for pattern discovery.

PCA

PCA extracts statistically independent features by finding a factorial representation of a signal distribution for linear-correlated and Gaussian-distributed signals. PCA transforms the original variables into new ones that are uncorrelated and account for decreasing proportions of the variance in the data. The aim of this method is to reduce the dimensionality of the data. The new variables, the principal components, are defined as linear functions of the original variables. If the first few principal components account for a large percentage of the variance, for example, greater than 70%, they can be used to both simplify subsequent analyses and display and summarize the data in a parsimonious manner.

PCA was used in a controlled experiment that randomized mice with cancer to either be administered or not be administered tamoxifen and identified genes with altered expression caused by tamoxifen [6]. This was the use of PCA in a subtraction task for dimension reduction rather than for pattern discovery. The primary problem with PCA for pattern discovery is that it does not consider the complex character of genes. Microarrays are being used because every gene is potentially informative, and many genes are nonlinearly and interactionally associated. Thus, the assumption that most genes are linear and noninteractional and possess uniform variance oversimplifies the problem and results in an ill-fitting solution. This technique of combining genes into large groups based on an overly simplistic algorithm provides little information about the disease process. In addition, PCA is sensitive to data transformations and outliers. Finally, one must consider the need to decorrelate the higher-order moments in the input. This can be accomplished by a generalization of PCA, namely, independent component analysis [9]. In other words, for complex phenomena, PCA is unable to separate signal from noise [10].

SOM Algorithms

SOMs were introduced by Kohonen [11] in 1984 as a tool for visualizing data. The SOM method has recently been proposed for microarray analysis [12–14]. In the SOM approach, the entire training data

Table 1. Statistical Methods for Analyzing Microarrays

Learning Methods	Problems Associated With Methods
Unsupervised	
Principal components analysis	Oversimplifies the pattern; causes the loss of all but the most obvious patterns
Self-organizing maps	Shown to perform poorly when there are many clusters
Clustering	Relies on distance measures, but different measures produce different results
Supervised	
Support vector machines	Not tested on a difficult problem
Classification and regression trees	Quickly runs out of data; no internally valid method for branching
Linear discriminant analysis	Assumes linearity among the variables, which is almost never true

set is used in each training iteration because batch processing is not affected by presentation order and is faster [15]. The batch SOM algorithm consists of two steps. First, the training data are partitioned in terms of their locations. Second, the units are updated by taking weighted centers of the data falling into the certain regions, with the weighting function given by the neighborhood. The neighborhood width is decreased, and steps 1 and 2 are repeated. The second step can be considered as a smoothing procedure (like a weighted average) [16]. A problem with this approach is that it does not optimize an objective function, and there may not be an objective function for the SOM algorithm [17]. Also, in some situations, neighborhood preservation is not guaranteed by the SOM procedure.

In a series of multivariate normal clustering problems, SOM was shown to perform significantly worse in recovering the structure of clusters and preserving the topology compared with more traditional methods [18]. One reason for the poor performance of SOMs is that the estimation error of its smoothers increases for a fixed sample size as the dimensionality increases [16]. In other words, a characteristic of SOMs is that their error increases with large numbers of clusters in the data [19].

Clustering Algorithms

Eisen et al. [20] and others [21–25] recently applied traditional hierarchical clustering methods to microarray analysis. This approach produces a hierarchical dendrogram in which genes with similar expression patterns, according to the standard correlation coefficient metric, are adjacent, and adjacency is interpreted as functional similarity. In this approach, a local criterion is used to build clusters by using the local structure of the data. The object of cluster analysis is to determine a classification or taxonomic scheme that accounts for the variance among subjects. The main issues with generic clustering algorithms are: (1) many different types of variance participate in the overall variance, and each variance component participates to a different degree depending on the variables, the data set, and the task; and (2) many equally plausible clustering models account for the variance, and no algorithm exists to allow us to separate the correct model from all the incorrect models.

Clustering, SOMs, and related unsupervised learning techniques rely on measures of similarity, i.e., distance measures that operate on feature vectors, usually Euclidean distance. Some of the problems with these metrics are: they are generic across problem domains, they are too weak to capture complex phenomena, and they present the false appearance of utility by capturing easily discovered relationships. The unsophisticated nature of clustering is exemplified by the most commonly used distance metric, namely, the Euclidean distance calculation, as follows:

For two observations:

$$x' = [x_1, \ldots, x_n] \text{ and } y' = [y_1, \ldots, y_n]$$

$$d_{xy} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

A major problem with clustering algorithms is that transformation invariance does not hold when generic distance metrics are used. In other words, different generic distance measures (Euclidean vs city block, etc) produce different cluster results. An additional issue is the lack of independence among gene locations and the resulting correlations. Also, clustering algorithms that minimize some function of scatter matrices do not have convergence proofs [26]. Finally, overlapping class structures from multiple causes are not easily accommodated by hierarchical clustering methods.

It would be better if there were domain-specific distance measures. However, this idea does not solve the problem; rather, it shifts the problem to how to select the optimal domain-specific distance metric. Creating a domain-specific distance measure can be dangerous because an incorrect metric will almost certainly result in biased and misleading results. In addition, it is very difficult to create a proper mathematical proof for a domain-specific metric.

Supervised Learning

In supervised learning tasks, the final error metrics are available during training; therefore, the algorithm can directly reduce the number of misclassifications in the training data set. It is usually a good strategy to try to turn an unsupervised learning problem into a supervised learning problem because more information can be brought to bear on the solution.

Brown et al. [27] used prior knowledge of gene function to identify unknown genes of similar func-

tion from expression data using a support vector machine (SVM) classification approach. An SVM performs mapping in high-dimensional space. In the separable case, the SVM algorithm constructs the separating hyperplane for which the margin between the positive and negative examples in space is maximized. It uses a function that can be viewed as the measure of similarity between genes. It is known that SVMs give good results on pattern recognition problems although they do not incorporate problem domain knowledge, but they can be very slow to converge [28]. Brown et al. [27] found that SVM performed more accurately than classification and decision trees (classification and regression trees), Parzen windows, and Fisher's linear discriminant. However, these three algorithms are easily defeated. The problem with supervised learning algorithms is that learning is best achieved when there are relatively few variables and many instances. Unfortunately, microarray data sets usually contain thousands of variables (genes) and only a few instances (patients) of each variable.

Sources of Error in the Analysis of Microarray Data

The following sources of error exist in microarray data. (1) The microarray technology itself; low levels of a messenger RNA are not generally detected on a microarray. Microarrays require large numbers of homogeneous cells to detect low-abundance messages. This can be a problem for screening applications and the early detection of disease. Such solutions as RT-PCR can create new problems. In addition, in some cases, the results of different microarray technologies cannot be combined because of differences in baseline conditions and the fact that the output of a microarray can be relative rather than absolute, even when the results are numeric. (2) The patients; the composition of the specimen that is used, including the ratio of normal to malignant cells; the patient's stage of disease process at discovery; and individual and population genetic variation. (3) The disease; there may be subtypes of a cancer site (e.g., within breast cancer) and different pathways that result in the final common pathway of clinical cancer. (4) The inefficiency of the statistical method in capturing the pattern. (5) Random error.

Pattern Validation

After a putative pattern is discovered, it must be validated. A common validation approach is the robust heuristic. This approach begins with the investigators examining a microarray and identifying a large number of interesting patterns. The investigators then turn to a second microarray (second patient) with the same disease and the same genes on the microarray. They look for the same or similar patterns in the second microarray that they found in the original microarray. Usually, they find several of the same patterns detected in the original microarray. This heuristic approach is continued for several more microarrays, finding fewer and fewer of the same patterns. When this process is completed, one or more of the original patterns will have occurred in all the microarrays. The claim is made that because a pattern occurred in all the microarrays, the pattern is robust. In other words, the claim is that it is unlikely that the observed pattern occurred in all the microarrays by chance.

Unfortunately, one cannot make such a claim using this approach. In a large microarray, many patterns occur by chance. If each pattern is a small number of genes, then by chance, some of the patterns will occur in several microarrays. This reality is obscured by the elimination of patterns as additional microarrays are sequentially searched. The correct approach is to hypothesize a putative pattern before seeing the data and test it on a large number of independently derived microarrays using very stringent criteria for testing whether the pattern occurred by chance. The problem with testing the pattern is that the error (variance) for each gene and each microarray is not known; thus, it is difficult to determine whether the observed similarity or difference between microarrays could have occurred by chance. Currently, it is very difficult to perform meaningful significance testing of microarrays.

Promising Areas of Investigation

There are several interesting approaches to microarray-based pattern discovery. Stochastic search methods may be useful for finding putative gene patterns [29,30]. These methods attempt to identify possible subsets of explanatory variables that can then be evaluated. Other approaches currently being assessed for their efficiency in dealing with mas-

sively parallel data include a combinatorial multivariate method [31] and mixture models [32–34]. The combinatorial method proposed by Califano [31] is a supervised learning task that improves as the number of cases increases. Finite mixture models assume that the data arise from a mixture of several unknown heterogeneous populations. Titterington et al. [35] note that finite mixture distributions have been used as models since the work of Newcomb [36] in 1886 and Pearson [37] in 1894. Mixture models can be used for unsupervised and learning tasks. Many estimation methods can be applied to finite mixture problems, including the method of moments, maximum likelihood, minimum chi-square, least-squares, and Bayesian approaches [35,38–40]. An important issue in the direct application of mixture models is how to determine the number of components and their distributions.

Strong Claim

Given the current state-of-the-art and the limited number of available microarrays, traditional statistical methods applied to the entire microarray are capable of finding only the grossest of patterns, those unlikely to lead to pharmacogenomic targets. In addition, there is currently no method to verify that a pattern found in a top-down analysis of a large microarray is true and not caused by chance.

Solutions

Four areas require improvement: (1) refining the microarray technology, (2) minimizing the patient and disease variance through careful data acquisition and sample preparation, (3) acquiring more cases, and (4) improving the statistical pattern detection algorithms. However, improvements in these areas will not solve the NP-hard problem or deal with our inability to determine whether the patterns we discover by top-down analysis are true.

The problem can be reformulated to be computationally tractable and statistically accessible. We begin with a known genetic pattern, i.e., a known gene expression relationship, and add genes. Using artificial neural networks and mixture models, we perform hypothesis-driven experiments by asking whether a gene on the microarray is related to the known gene relationship expressed in the microarray. In addition, we can order the genes on the array in a manner we believe to be consistent with this true relationship.

Before microarrays, we would have had to go to our laboratory and, after several months of bench research, come to some conclusion regarding the relationship of that gene to the known gene pattern. With the microarray, we can perform an instant experiment by examining the expression of the gene of interest on many chips (patients) in relation to the known gene relationship also expressed on those chips. Microarrays allow us to immediately test our experimental hypotheses in the context of the other relevant genes. Thus, the more genes in the array, the better. The optimal microarray presents the expression of all the genes in the human genome.

Disease Modeling and Simulation

Fully representing a disease process at the proteogenomic level, even with the use of microarray tools, will require many years of rigorous research. Before a complete description of a disease, we will want to make use of the data we have acquired with our new powerful molecular genetic tools, i.e., use incomplete information to benefit patients. To use fragmentary proteogenomic information for pharmacogenomic ends, we need to create mathematical models of the disease process. We will then be in a position to perform simulation studies to fill in what we do not know. Modeling allows us to extend our knowledge beyond the data and provides reasonable estimates of the phenomena, which in turn allow us to identify therapy targets. We can use the model to integrate known disease components and simulate the disease process.

The accuracy of the simulation depends on: (1) the known components of the disease process (the more we know, the better the model), (2) the accuracy of our knowledge (the more correct the data, the better the model), and (3) the adequacy of the mathematical method used to model and simulate the disease. The adequacy of the model rests on its assumptions and its ability to efficiently represent the disease (i.e., to fit the disease). Two additional benefits of modeling and simulation are that (1) it allows the assessment of the correctness of the data and the methods by experimentally testing the model's predictions [41], and (2) it allows assessment of

the correctness of new information by determining how well the information fits in our disease model.

Received March 20, 2000.
Received in revised form July 10, 2000.
Accepted August 15, 2000.

References

1. Cooper DL: The promise and the dilemma of the new millennium. Mol Diagn 2000;5:7–8
2. Burke HB, Henson DE: Evaluating prognostic factors. CME J Gynecol Oncol 1999;4:244–252
3. Lander ES: Array of hope. Nat Genet 1999;21S:3–4
4. Sehgal A, Boynton AL, Yong RF, et al.: Applications of the differential hybridization of Atlas Human expression arrays technique in the identification of differentially expressed genes in human glioblastoma tumor tissue. J Surg Oncol 1998;67:234–241
5. Shim C, Shang W, Rhee CH, Lee JH: Profiling of differentially expressed genes in human primary cervical cancer by complementary DNA expression array. Clin Cancer Res 1998;4:3045–3050
6. Hilsenbeck SG, Friedrichs WE, Schiff R, et al.: Statistical analysis of array expression data as applied to the problem of tamoxifen resistance. J Natl Cancer Inst 1999;91:453–459
7. Khan J, Simon R, Bittner M, et al.: Gene expression profiling of alveolar rhabdomyosarcoma with cDNA microarrays. Cancer Res 1998;58:5009–5013
8. Lee C, Klopp RG, Weindruch R, et al.: Gene expression profile of aging and its retardation by caloric restriction. Science 1999;285:1390–1393
9. Comon P: Independent component analysis—A new concept? Signal Processing 1994;36:287–314
10. Wittis J, Friedman H: Searching for evidence of altered gene expression: A comment on statistical analyses of microarray data. J Natl Cancer Inst 1999; 91:400–401
11. Kohonen T: Self-organizing maps, 2nd ed. Springer, Heidelberg, 1997
12. Kohonen J, Bubenddorf L, Kallioniemi A, et al.: Tissue microarrays for high-throughput molecular profiling of tumor specimens. Nat Med 1998;4:844–847
13. Tamayo P, Slonim D, Mesirov J, et al.: Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. Proc Natl Acad Sci U S A 1999;96: 2907–2912
14. Toronen P, Kolehmainen M, Wong C, Castren E: Analysis of gene expression data using self-organizing maps. FEBS Lett 1999;451:142–146
15. Kohonen T: Things you haven't heard about the self-organizing map. Proc IEEE Int Joint Conf Neural Networks, San Francisco, 1993, pp. 1147–1156
16. Mulier F, Cherkassy V: Self-organization as an iterative kernel smoothing process. Neural Comput 1995;7:1165–1177
17. Erwin E, Obermayer K, Schulten K: Self-organizing maps: Ordering, convergence properties and energy functions. Biol Cybern 1992;67:47–55
18. Flexer A: Limitations of self-organizing maps for vector quantization and multidimensional scaling. In Mozer MC, Jordan MI, Petsche T: Advances in neural information processing systems 9. MIT Press, Cambridge, MA, 1997, pp. 445–451
19. Bezdek JC, Nikhil RP: An index of topological preservation for feature extraction. Pattern Recognition 1995;28:381–391
20. Eisen MB, Spellman PT, Brown PO, Botstein D: Cluster analysis and display of genome-wide expression patterns. Proc Natl Acad Sci U S A 1998;95: 14863–14868
21. Perou CM, Jeffrey SS, van de Rijn M, et al.: Distinctive gene expression patterns in human mammary epithelial cells and breast cancer. Proc Natl Acad Sci U S A 1999;96:9212–9217
22. Iyer VR, Eisen MB, Ross DT, et al.: The transcriptional program in the response of human fibroblasts to serum. Science 1999;283:83–87
23. Alizadeh AA, Eisen MB, Davis RE, et al.: Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature 2000;403:503–511
24. Michaels GS, Carr DB, Askenazi M, Fuhrman S, Wen X, Somogyi R: Cluster analysis and data visualization of large-scale gene expression data. Pac Symp Biocomput 1998;4:42–53
25. Wen X, Fuhrman S, Michaels GS, et al.: Large-scale temporal gene expression mapping of central nervous system development. Proc Natl Acad Sci U S A 1998;95:334–339
26. Fukunaga K: Statistical pattern recognition. Academic Press, New York, 1990
27. Brown MPS, Grundy WN, Lin D, et al.: Support vector machine classification of microarray gene expression data. University of California Technical Report USCC-CRL-99-09, 1999. Available at: http://www.cse.ucsc.edu/research/compbio/genex. Accessed: November 5, 1999
28. LeCun Y, Jackel L, Bottou L, et al.: Comparison of learning algorithms for handwritten digit recognition. In Fogelman F, Gallinari P (eds): International Conf Artificial Neural Networks, Lawrence Erlbaum, Hillsdale, NJ, 1995, pp. 53–60
29. George E, McCulloch R: Variable selection via Gibbs sampling. J Am Stat Assoc 1993;88:881–889
30. George E, McCulloch R: Stochastic search variable

selection. In Gilks WR, Richardson S, Spiegelhalter DJ: Markov chain Monte Carlo in practice. Chapman & Hall, London, 1996, chapter 12, pp 203–213
31. Califano A: SPLASH: Structural pattern localization algorithm by sequence histograming. Bioinformatics 2000;16:341–357
32. Wallace CS, Dowe DL: MML mixture modelling of multi-state, Poisson, von Mises circular and Gaussian distributions. Proceedings Sixth International Workshop on Artificial Intelligence and Statistics, Ft Lauderdale, FL, January 4–7, 1997, pp. 529–536
33. Fraley C, Raftery AE: How many clusters? Which clustering method?—Answers via model-based cluster analysis. Comput J 1998;41:578–588
34. Liu JS, Neuwald AF, Lawrence CE: Markovian structures in biological sequence alignment. J Am Stat Assoc 1999;94:1–15
35. Titterington DM, Smith AFM, Makov UE: Statistical analysis of finite mixture distributions. Wiley, Chichester, 1985
36. Newcomb S: A generalized theory of the combination of observations so as to obtain the best result. Am J Math 1886;8:343–366
37. Pearson K: Contributions to the mathematical theory of evolution. Philos Trans R Soc 1894;185:71–110
38. Roeder K: A graphical technique for determining the number of components in a mixture of normals. J Am Stat Assoc 1994;89:487–495
39. Richardson S, Green PJ: On Bayesian analysis of mixtures with an unknown number of components. JR Stat Soc: B 1997;59:731–792
40. Bohning D: Computer-assisted analysis of mixtures and applications. Chapman & Hall, London, 1999
41. Fussenegger M, Bailey JE, Varner J: A mathematical model of caspase function in apoptosis. Nat Biotech 2000;18:768–774