CHAPTER 14

TECHNIQUES FOR ANALYSIS OF GENE EXPRESSION DATA

Jinyan Li
Institute for Infocomm Research
jinyan@i2r.a-star.edu.sg

Limsoon Wong
Institute for Infocomm Research
limsoon@i2r.a-star.edu.sg

The development of microarray technology in the last decade has made possible the simultaneous monitoring of the expression of thousands of genes. This development offers great opportunities in advancing the diagnosis of dieases, the treatment of diseases, and the understanding of gene functions. This chapter provides an in-depth survey of several approaches to some of the gene expression analysis challenges that accompany these opportunities.

ORGANIZATION.

Section 1. We begin with a brief introduction to microarrays in terms of how they are made and how they are used for measuring the expression level of thousands of genes simultaneosly.

Section 2. Then we discuss how to diagnose disease subtypes and states by microarray gene expression analysis. We present a standard approach that combines gene selection and subsequent machine learning. Besides applying gene selection and machine learning methods from Chapter 3, we also present the shrunken centroid method of Tibshirani *et al.* in some detail. The subtype diagnosis of childhood acute lymphoblastic leukaemia is used as an example.

Section 3. Next we consider the problem of discovering new disease subtypes by means of microarray gene expression analysis. We relate the discovery of a novel subtype of childhood acute lymphoblastic leukaemia via a hierarchical clustering approach. We also describe the discovery of novel transcription factor binding sites and novel gene funtional groupings via a fuzzy k-means clustering approach.

Section 4. Then we look at the problem of infering how the expression of one gene influences the expression of another gene. We present two approaches; one is based on

mining of association rules, the other is based on an ingenious use of normal classifiers. We also describe the concept of interaction generality in the related problem of detecting false positives from high-throughput protein-protein interaction experiments.

Section 5. Lastly, we present a highly speculative use of gene expression patterns to formulate treatment plans.

1. Microarray and Gene Expression

Microarrays or DNA chips are powerful tools for analyzing the expression profiles of gene transcripts under various conditions.[518, 890] These microarrays contain thousands of spots of either cDNA fragments corresponding to each gene or short synthetic oligonucleotide sequences. By hybridizing labeled mRNA or cDNA from a sample to a microarray, transcripts from all expressed genes can be assayed simultaneously. Thus one microarray experiment can yield as much information as thousands of Northern blots. It is hopeful that better diagnosis methods, better understanding of disease mechanisms, and better understanding of biological processes, can be derived from a careful analysis of microarray measurements of gene expression profiles.

There are two main types of microarray. The first type is based on the scheme of Fodor *et al.*[253] that uses lithographic production techniques to synthesize an array of short DNA fragments called oligos. Here is a brief outline of their scheme. First a silicon surface is coated with linker molecules that bind the four DNA building blocks, adenine (A), cytosine (C), guanine (G), and thymine (T). These linkers are initially capped by a "blocking" compound that can be removed by exposure to light. By shining light through a mask, those areas of the silicon surface that correspond to holes in the mask become exposed. The chip is then incubated with one of the four DNA building blocks, say adenine, which then binds to those exposed areas. After that, the blocking compound is reapplied. By repeating this process with different masks and different DNA building blocks, an array of different oligos can be built up easily, as shown in Figure 1.

Each oligo can bind stretches of DNA that have complementary sequences to the oligo in the usual Crick-Watson way. Then the following procedure is followed to use the microarray to monitor the expression of multiple genes in a sample. RNAs are isolated from samples, converted into cDNAs, and conjugated to biotin. These biotin-conjugated cDNAs are then fragmented by heat, and hybridized with the oligos on the microarray. A washing step then follows to get rid of unbound cDNAs. The strands that are bound to the microarray can then be stained by a streptavidin-linked fluorescent dye, and detected by exciting the fluorescent tags with a laser. Since the sequence of each oligo on the microarray is known by construction, it is easy to know the sequence of the cDNA that is bound to a

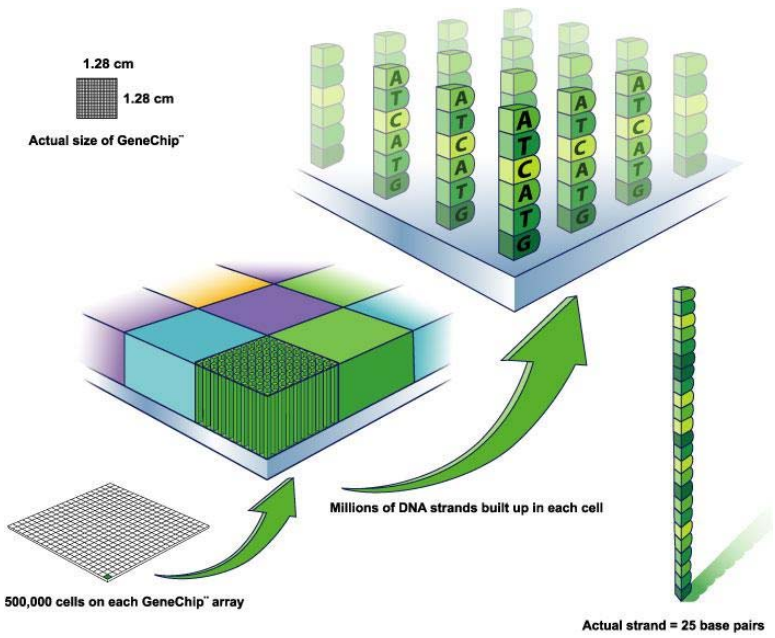

Fig. 1. A cartoon of the oligos on a microarray. Notice that the sequence of each oligo and their position on the microarray are known by construction. The oligo-based microarrays made by Affymetrix are called GeneChip® microarrays, and the oligos are 25 nucleotides in length. (*Image credit: Affymetrix*.)

particular position of the microarray, as shown in Figure 2.

The second popular type of microarrays is based on the scheme developed at Stanford.[752,770] Here, cDNAs are directly spotted onto a glass slide, which is treated with chemicals and heat to attach the DNA sequences to the glass surface and denature them. This type of microarray is primarily used for determining the relative level of expression of genes in two contrasting samples. The procedure is as follows. The fluorescent probes are prepared from two different mRNA sources with the use of reverse transcriptase in the presence of two different fluorophores. The two set of probes are then mixed together in equal proportions, hybridized to a single array, and scanned to detect fluorescent color emissions corresponding to the two fluorophores after independent excitation of the two fluorophores. The differential gene expression is then typically calculated as a ratio of these two fluorescent color emissions.

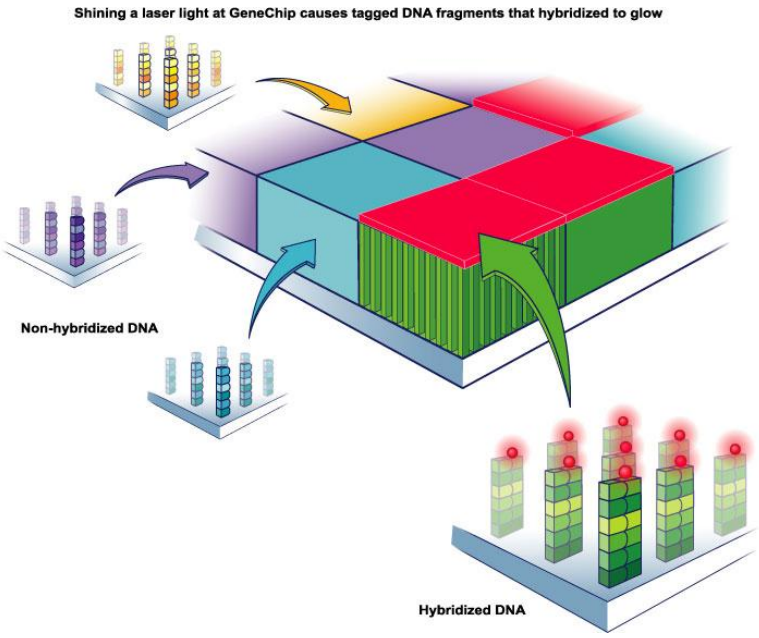

Fig. 2. A cartoon depicting scanning of tagged and un-tagged probes on an Affymetrix GeneChip® microarray. (*Image credit: Affymetrix.*)

2. Diagnosis by Gene Expression

A single microarray experiment can measure the expression level of tens of thousands of genes simultaneously.[518, 698] In other words, the microarray experiment record of a patient sample—see Figure 3 for an example—is a record having tens of thousands of features or dimensions. A major excitement due to microarrays in the biomedical world is the possibility of using microarrays to diagnose disease states or disease subtypes in a way that is more efficient and more effective than conventional techniques.[20, 269, 297, 663, 918]

Let us consider the diagnosis of childhood leukaemia subtypes as an illustration. Childhood leukaemia is a heterogeneous disease comprising more than 10 subtypes, including T-ALL, E2A-PBX1, TEL-AML1, BCR-ABL, MLL, Hyperdiploid>50, and so on. The response of each subtype to chemotherapy is different. Thus the optimal treatment plan for childhood leukaemia depends critically on the subtype.[686] Conventional childhood leukaemia subtype diagnosis is a difficult and expensive process.[918] It requires intensive laboratory studies comprising cytogenetics, immunophenotyping, and molecular diagnostics. Usually,

probe	pos	neg	pairs in avg	avg diff	abs call	Description
...	...	...	...	...	...	...
107_at	4	4	15	3723.3	A	Z95624 Human DNA ...
108_g_at	5	2	15	1392.4	A	Z95624 Human DNA ...
109_at	6	2	16	2274.7	M	Z97074 Human mRNA ...
...	...	...	...	...	...	...

Fig. 3. A partial example of a processed microarray measurement record of a patient sample using the Affymetrix® GeneChip® U95A array set. Each row represents a probe. Typically each probe represents a gene. The U95A array set contains more than 12,000 probes. The 5th column contains the gene expression measured by the corresponding probe. The 2nd, 3rd, 4th, and 6th columns are quality control data. The 1st and last columns are the probe identifier and a description of the corresponding gene.

these diagnostic approaches require the collective expertise of a number of professionals comprising hematologists, oncologists, pathologists, and cytogeneticists. Although such combined expertise is available in major medical centers in developed countries, it is generally unavailable in less developed countries. It is thus very exciting if microarrays and associated automatic gene expression profile analysis can serve as a single easy-to-use platform for subtyping of childhood ALL.

2.1. *The Two-Step Approach*

The analysis of gene expression profiles for the diagnosis of disease subtypes or states generally follows a two-step procedure first advocated by Golub *et al.*,[297] *viz.*

(1) selecting relevant genes and
(2) training a decision model using these genes.

The step of selecting relevant genes can be performed using any good feature selection methods such as those presented in Chapter 3—signal-to-noise measure,[297] t-test statistical measure,[133] entropy measure,[242] χ^2 measure,[514] information gain measure,[692] information gain ratio,[693] Fisher criterion score,[251] Wilcoxon rank sum test,[742] principal component analysis,[399] and so on. The step of decision model construction can be performed using any good ma-

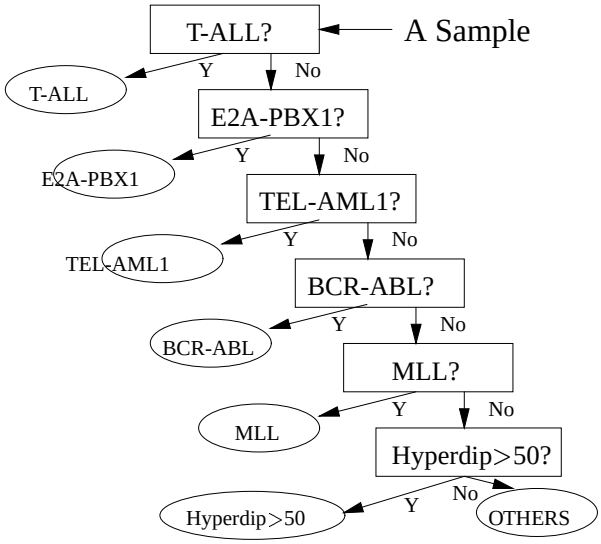

Fig. 4. The classification of the ALL subtypes is organized in a tree. Given a new sample, we first check if it is T-ALL. If it is not classified as T-ALL, we go to the next level and check if it is an E2A-PBX1. If it is not classified as E2A-PBX1, we go to the third level and so on.

chine learning methods such as those presented in Chapter 3—decision tree induction methods,[693] Bayesian methods,[214] support vector machines (SVM),[855] PCL,[492, 497] and so on.

We illustrate this two-step procedure using the childhood acute lymphoblastic leakaemia (ALL) dataset reported in Yeoh *et al.*[918] The whole dataset consists of gene expression profiles of 327 ALL samples. These profiles were obtained by hybridization on the Affymetrix® GeneChip® U95A array set containing probes for 12558 genes. The data contain all the known acute lymphoblastic leukaemia subtypes, including T-ALL, E2A-PBX1, TEL-AML1, BCR-ABL, MLL, and Hyperdiploid>50. The data are divided by Yeoh *et al.* into a training set of 215 instances and an independent test set of 112 samples. The original training and test data are layered in a tree-structure, as shown in Figure 4. Given a new sample, we first check if it is T-ALL. If it is not classified as T-ALL, we go to the next level and check if it is an E2A-PBX1. If it is not classified as E2A-PBX1, we go to the third level and so on.

Li *et al.*[492] are the first to study this dataset. At each level of the tree, they first use the entropy measure[242] and the χ^2 measure[514] to select the 20 genes that are most discriminative in that level's training data. Then they apply the PCL

Testing Data	Error rate of different models			
	C4.5	SVM	NB	PCL
T-ALL vs OTHERS1	0:1	0:0	0:0	0:0
E2A-PBX1 vs OTHERS2	0:0	0:0	0:0	0:0
TEL-AML1 vs OTHERS3	1:1	0:1	0:1	1:0
BCR-ABL vs OTHERS4	2:0	3:0	1:4	2:0
MLL vs OTHERS5	0:1	0:0	0:0	0:0
Hyperdiploid>50 vs OTHERS	2:6	0:2	0:2	0:1
Total Errors	14	6	8	4

Fig. 5. The error counts of various classification methods on the blinded ALL test samples are given in this figure. PCL is shown to make considerably less misclassifications. The OTHERSi class contains all those subtypes of ALL below the ith level of the tree depicted in Figure 4.

classifier[492] on the training data using those 20 genes to construct a decision model to predict the subtypes of test instances of that level. The entropy measure, the χ^2 measure, and the PCL classifier are described in Chapter 3.

For comparison, Li *et al.* have also applied several popular classification methods described in Chapter 3—C4.5,[693] SVM,[855] and Naive Bayes (NB)[214]—to the same datasets after filtering using the same selected genes. In each of these comparison methods, the default settings of the weka package are used. The weka package can be obtained at `http://www.cs.waikato.ac.nz/ml/weka`.

The number of false predictions on the test instances, after filtering by selecting relevant genes as described above, at each level of the tree by PCL, as well as those by C4.5, SVM, and NB, are given in Figure 5. The results of the same algorithms but without filtering by selecting relevant genes beforehand are given in Figure 6, which clearly shows the beneficial impact of the step of gene selection.

2.2. *Shrunken Centroid Approach*

Tibshirani *et al.*[827] also use the same two-step approach to diagnose cancer type based on microarray data. However, the details of their approach are different from those basic methods already described in Chapter 3. Their approach performs well, is easy to understand, and is suitable for the situation where there are more than two classes.

Testing Data	Error rate of different models		
	C4.5	SVM	NB
T-ALL vs OTHERS1	0:1	0:0	13:0
E2A-PBX1 vs OTHERS2	0:0	0:0	9:0
TEL-AML1 vs OTHERS3	2:4	0:9	20:0
BCR-ABL vs OTHERS4	1:3	2:0	6:0
MLL vs OTHERS5	0:1	0:0	6:0
Hyperdiploid>50 vs OTHERS	4:10	12:0	7:2
Total Errors	26	23	63

Fig. 6. The error counts of various classification methods on the blinded ALL test samples without filtering by selecting relevant genes are given in this figure. The OTHERSi class contains all those subtypes of ALL below the ith level of the tree depicted in Figure 4.

The gist of the approach of Tibshirani *et al.* is as follows. Let X be a matrix of gene expression values for p genes and n samples. Let us write $X[i, j]$ for the expression of gene i in sample j. Suppose we have k classes and we write C_h for the indices of the n_h samples in class h. We create a "prototype" gene expression profile vector Y_h, also called a "shrunken" centroid, for each class h. Then given a new test sample t, we simply assign to it the label of the class whose prototype gene expression profile is closest to this test sample.

Let us use the notations $\langle e_i \mid i = 1 \ldots n \rangle$ to mean the vector $\langle e_1, \ldots, e_n \rangle$. Then the usual centroid for a class h is the vector

$$Z_h = \left\langle \frac{\sum_{j \in C_h} X[i,j]}{n_h} \;\middle|\; i = 1 \ldots p \right\rangle$$

where the ith component $Z_h[i]$ is the mean expression value in class h for gene i. And the overall centroid is a vector O where the ith component $O[i] = \sum_{j=1}^{n} X[i, j]/n$ is the mean expression value of gene i over samples in all classes.

In order to give higher weight to genes whose expression is stable within samples of the same class, let us standardize the centroid of each gene i by the within-class standard deviation in the usual way, *viz.*

$$d_{ih} = \frac{Z_h[i] - O[i]}{\sqrt{\frac{1}{n_h} + \frac{1}{n}} \times (s_i + s_0)}$$

where s_i is the pooled within-class standard deviation for gene i, *viz.*

$$s_i^2 = \frac{1}{n-k} \times \sum_h \sum_{j \in C_h} (X[i,j] - Z_h[i])^2$$

and the value s_0 is a positive constant—with the same value for all genes—included to guard against the possibility of large d_{ih} values arising by chance from genes with low expression levels. Tibshirani *et al.* suggest to set s_0 to the median value of s_i over the set of genes.

Thus d_{ih} is a t-statistics for gene i, comparing class h to the overall centroid. We can rearrange the equation as

$$Z_h[i] = O[i] + \sqrt{\frac{1}{n_h} + \frac{1}{n}} \times (s_i + s_0) \times d_{ih}$$

Tibshirani *et al.* shrink d_{ih} toward zero, giving d'_{ih} and yielding the shrunken centroid or prototype Y_h for class h, where

$$Y_h[i] = O[i] + \sqrt{\frac{1}{n_h} + \frac{1}{n}} \times (s_i + s_0) \times d'_{ih}$$

The shrinkage they use is a soft thresholding: each d_{ih} is reduced by an amount Δ in absolute value and is set to 0 if it becomes less than 0. That is,

$$d'_{ih} = \begin{cases} sign(d_{ih})(|d_{ih}| - \Delta) & \text{if } |d_{ih}| - \Delta > 0 \\ 0 & \text{otherwise} \end{cases}$$

Because many of the $Z_h[i]$ values are noisy and close to the overall mean $O[i]$, soft thresholding usually produces more reliable estimates of the true means. This method has the advantage that many of the genes are eliminated from the class prediction as the shrinkage parameter Δ is increased. To see this, suppose Δ is such that $d'_{ih} = 0$. Then the shrunken centroid $Y_h[i]$ for gene i for any class h is $O[i]$, which is independent of h. Thus gene i does not contribute to the nearest shrunken centroid computation. By the way, Δ is normally chosen by cross-validation.

Let t be a new sample to be classified. Let $t[i]$ be the expression of gene i in this sample. We classify t to the nearest shrunken centroid, standardizing by $s_i + s_0$ and correcting for class population biases. That is, for each class h, we first compute

$$\delta_h(t) = \sum_{i=1}^{p} \left(\frac{t[i] - Y_h[i]}{s_i + s_0} \right)^2 - 2 \times \log(\pi_h)$$

The first term here is simply the standardized squared distance of t to the hth shrunken centroid. The second term here is a correction based on the class prior probability π_h, which gives the overall frequency of class h in the population, and

is usually estimated by $\pi_h = n_h/n$. Then we assign t the label h that minimizes $\delta_h(t)$. In other words, the classification rule is

$$C(t) = \alpha, \text{ where } \delta_\alpha(t) = \min_h \delta_h(t)$$

This approach appears to be very effective. On the acute lymphoblastic leukaemia dataset of Golub *et al.*,[297] Tibshirani *et al.*[827] report that at $\Delta = 4.06$—the point at which cross-validation error starts to rise quickly—yields 21 genes as relevant. Furthermore, these 21 genes produce a test error rate of 2/34. In comparison, Golub *et al.*[297] use 50 genes to obtain a test error rate of 4/34. On the childhood small round blue cell tumours dataset of Khan *et al.*,[430] Tibshirani *et al.* report that at $\Delta = 4.34$ there are 43 genes that are relevant. Furthermore, these 43 genes produce a test error rate of 0. This result is superior to that of Khan *et al.*, who need 96 genes to achieve the same test error rate.

3. Co-Regulation of Gene Expression

In the preceding section we see that it is possible to diagnose disease subtypes and states from gene expression data. In those studies, we assume that all the disease subtypes are known. However, in real life, it is possible for a heterogeneous disease to have or to evolve new subtypes that are not previously known. Can computational analysis of gene expression data help uncover such new disease subtypes? Similarly, there are still many genes and their products whose functions are unknown. Can computational analysis of gene expression data help uncover functionally related gene groups? and can we infer the functions and regulation of such gene groups? Unsupervised machine learning methods, especially clustering algorithms, are useful for these problems. This section present two examples.

3.1. *Hierarchical Clustering Approach*

Let us use the childhood ALL dataset from Yeoh *et al.*[918] from Subsection 2.1 for illustration. As mentioned earlier, childhood ALL is a heterogeneous disease comprising 6 known major subtypes, *viz.* T-ALL, hyperdiploid with > 50 chromosomes, BCR-ABL, E2A-PBX1, TEL-AML1, and MLL gene rearrangements. However, the dataset from Yeoh *et al.* also contain some samples that are not assigned to any of these subtypes—these are the group marked as "OTHERS" in Figure 4.

This "OTHERS" group presents an opportunity for identifying new subtypes of childhood ALL. To do so, Yeoh *et al.*[918] perform a hierarchical clustering on their 327 childhood ALL samples using all the 12558 genes measured on their Affymetrix® GeneChip® U95A array set and using Pearson correlation as the

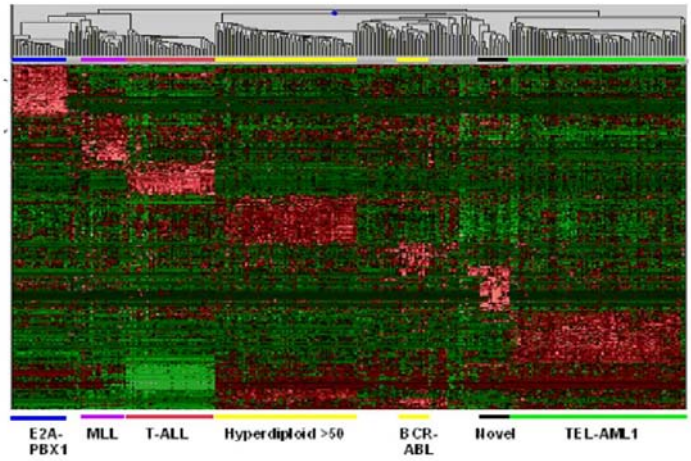

Fig. 7. Hierarchical clustering of 327 childhood ALL samples and genes chosen by χ^2 measure. Each column represents a sample, each row represents a gene. Note the 14 cases of the novel subtype.

distance between samples. Remarkably, this analysis clearly identifies the 6 major childhood ALL subtypes mentioned above. Moreover, within the "OTHERS" group, a novel subgroup of 14 cases are identified that have a distinct gene expression profile. These 14 cases have normal, pseudodiploid, or hyperdiploid karyotypes, and lack any consistent cytogenetic abnormality.

Figure 7 depicts the result of a hierarchical clustering of the 327 childhood ALL samples. To improve visualization clarity, instead of presenting a clustering involving all 12558 genes, only the top 40 genes selected using the χ^2 measure for each of the 6 major groups and the novel group are retained in this figure. The 14 cases of the novel subtype is clearly visible.

Thus, clustering algorithms can be used to discover new disease subtypes and states. As an introduction to hierarchical cluster algorithms and to the χ^2 measure can be found in Chapter 3, we omit them in this chapter. The definition of Pearson correlation is given in the next subsection—however, for the current subsection, the G and H in that formula should be interpreted as vectors representing the expression values of genes in sample g and sample h, and thus $G[i]$ is the expression of gene i in sample g and $H[i]$ is the expression of gene i in sample h.

3.2. *Fuzzy K-Means Approach*

Gasch and Eisen[277] use a technique called fuzzy k-means[80] to cluster a large collection of gene expression data obtained under a variety of experimental conditions. The dataset comprises 6153 genes in 93 microarray experiments taken from genomic expression data of wild-type *S. cerevisiae* responding to zinc starvation,[532] phosphate limitation,[625] DNA damaging agents,[275] and stressful environmental changes.[276]

They have obtained several very interesting results from analysing the resulting clusters. First, they have identified some meaningful clusters of genes that hierarchical and standard k-means clustering methods are unable to identify. Second, many of their clusters that correspond to previously recognized groups of functionally-related genes are more comprehensive than those clusters produced by hierarchical and standard k-means clustering methods. Third, they are able to assign many genes to multiple clusters, revealing distinct aspects of their function and regulation. Forth, they have also applied the motif-finding algorithm MEME[39] to the promoter regions of genes in some of the clusters to find short patterns of 6 nucleotides that are over represented and thus identified a few potentially novel transcription factor binding sites.

Before we proceed to describe the fuzzy k-means clustering method, let us first fix some notations. Let X be a matrix of gene expression values of $|X|^r$ genes under a variety of $|X|^c$ conditions. We write $X[g,i]$ for the expression of gene g in condition i in X. We write $X[g,\_]$ for the vector $\langle X[g,i] \mid i = 1 \ldots |X|^c \rangle$ representing the expression pattern of gene g in all the conditions in X. We write $X[\_,i]$ for the vector $\langle X[g,i] \mid g = 1 \ldots |X|^r \rangle$ representing the expression of genes in condition i in X. Similar notations are used for other two dimensional matrices. Also, for a vector G, we write $|G|$ for its size and $G[j]$ for its jth element.

For any two vectors G and H of gene expression patterns of genes g and h over the same conditions, so that $G[i]$ is the expression of gene g in condition i and $H[i]$ is the expression of gene h in condition i, the Pearson correlation coefficient of the observations of G and H is defined as:

$$S(G,H) = \frac{1}{|G|} \times \sum_{i=1}^{|G|} \frac{G[i] - \mu_G}{\sigma_G} \times \frac{H[i] - \mu_H}{\sigma_H}$$

where μ_G and μ_H are the mean of observations on G and H, and σ_G and σ_H are the standard deviation of G and H:

$$\mu_G = \sum_{i=1}^{|G|} \frac{G[i]}{|G|} \quad \text{and} \quad \sigma_G = \sqrt{\sum_{i=1}^{|G|} \frac{(G[i] - \mu_G)^2}{|G|}}$$

and similarly for μ_H and σ_H. The corresponding Pearson distance $D(G,H)$ is defined as $1 - S(G,H)$.

Let V be a matrix representing $|V|^r$ cluster centroids of averaged gene expression values in $|V|^c$ conditions. The fuzzy k-means algorithm[80] is based on the minimization of the objective function below, for a given fuzzy partition of the dataset X into $|V|^r$ clusters having centroids V:

$$J(X,V) = \sum_{g=1}^{|X|^r} \sum_{j=1}^{|V|^r} M(X[g,\_],V[j,\_],V)^2 \times D(X[g,\_],V[j,\_])^2$$

where $M(X[g,\_],V[j,\_],V)$ is the membership of gene g in cluster j.

The cluster membership function is a continuous variable from 0 to 1. Its value is to be interpreted as the strength of a gene's membership in a particular cluster. That is, under fuzzy k-means, a gene can belong to several clusters. The cluster membership function is defined as:

$$M(X[g,\_],V[j,\_],V) = \frac{1}{D(X[g,\_],V[j,\_])^2} \Bigg/ \sum_{j=1}^{|V|^r} \frac{1}{D(X[g,\_],V[j,\_])^2}$$

During a cycle of fuzzy k-means clustering, the centroids are refined repeatedly. A centroid $V[j,\_]$ is refined to $V[j,\_]'$ on the basis of the weighted means of all the gene expression patterns in the dataset X according to

$$V[j,\_]' = \left\langle \frac{\sum_{g=1}^{|X|^r} M(X[g,\_],V[j,\_],V)^2 \times W(X,g) \times X[g,i]}{\sum_{g=1}^{|X|^r} M(X[g,\_],V[j,\_],V)^2 \times W(X,g)} \;\middle|\; i = 1 \ldots |V|^c \right\rangle$$

where the gene weight $W(X,g)$ is defined empirically as

$$W(X,g) = \left(\sum_{h=1}^{|X|^r} \frac{S(X[g,\_],X[h,\_]) - C}{1-C} \right)^2$$

and C is a correlation cutoff threshold. In the work of Gasch and Eisen,[277] they set $C = 0.6$.

In each clustering cycle, the centroids are iteratively refined until the average change in gene memberships between interations is < 0.001. After each clustering cycle, the centroids are combined with those identified in previous cycles, and replicate centroids are averaged as follows. Each centroid is compared to all other centroids in the set, and centroid pairs that are Pearson correlated at > 0.9 are replaced by the average of the two vectors. The new vector is compared to the remaining centroids in the set and is again averaged with those to which it is Pearson correlated at > 0.9. This process continues until each centroid is compared to all other existing centroids in the set.

At the end of a clustering cycle, those genes with a Pearson correlation at > 0.7 to any of the identified centroids are taken as belonging to the respective clusters. These genes are then removed from further consideration. The next cycle of fuzzy k-means clustering are carried out on the remaining genes—*i.e.*, those with Pearson correlation at ≤ 0.7 to all the centroids. Incidentally, by considering a gene whose Pearson correlation to a centroid is > 0.7 as belong to the cluster of that centroid, it is therefore possible for a gene to belong simultaneously to multiple clusters. This is a great advantage of the fuzzy k-means method over other clustering methods that do not allow a gene to belong to more than one cluster. The reason is that many genes in real life do have multiple functional roles and thus naturally should belong to multiple clusters.

Gasch and Eisen[277] perform 3 successive cycles of fuzzy k-means clustering. Since k clusters are desired at the end of the 3 cycles, they aim to produce $k/3$ clusters in each cycle. The first cycle of clustering is initialized by using the top $k/3$ eigen vectors from a principle component analysis[399] on their dataset as prototype centroids for that clustering cycle. Subsequent cycles of clustering are initialized similarly, except that principle component analysis is performed on the respective data subset used in that clustering cycle. As details of principle component analysis have already been described in Chapter 3, we do not repeat here.

4. Inference of Gene Networks

A large number of genes can be differentially expressed in a microarray experiment. Such genes can serve as markers of the different classes—such as tumour vs. normal—of samples in the experiment. Some of these genes can even be the primary cause of a sample being tumour. In order to decide which gene is part of the primary cause and which gene is merely a down-stream effect, the underlying molecular network has to be assembled and considered. After the causal genes are identified, we may want to further develop drug substances to target them. The two major causes of treatment failure by drug substances are side effects and compensation effects. Side effects arise because genes and their protein products other than the intended ones are also modulated by the drug substances in unexpected ways. Compensation effects arise due to existence of parallel pathways that perform similar functions of the genes and proteins targeted by the drug substances and these parallel pathways are not affected by those drug substances. An understanding of the underlying molecular network is also useful for suggesting how best to target the causal genes. Motivated by these reasons, construction of a database of molecular network on the basis of microarray gene expression experiments has been attempted.

Let us recall that in analysing microarray gene expression output in the last two sections, we first identify a number of candidate genes by feature selection. Do we know which ones of these are causal genes and which are mere surrogates? Genes are "connected" in a "circuit" or network. The expression of a gene in a network depends on the expression of some other genes in the network. Can we reconstruct the gene network from gene expression data? For each gene in the network, can we determine which genes affect it? and how they affect it—positively, negatively, or in more complicated ways? There are several techniques to reconstructing and modeling molecular networks from gene expression experiments. Some techniques that have been tried are Bayesian networks,[263] Boolean networks,[16,17] differential equations,[153] association rule discovery,[643] classification-based methods,[783] and several other approaches to related problems.[380,734,735]

We devote the rest of this section to describe the classification-based method of Soinov *et al.*,[783] the association rules method of Creighton and Hanash,[176] and the interaction generality method of Saito *et al.*[734] The last method—interaction generality[734,735]—is actually concerned more with assessing the reliability of protein-protein interaction networks than with gene networks. However, it has been shown[280,301,390] that the average correlation coefficient of gene expression profiles that correspond to interacting gene products is higher than that of random pairs of gene products. Therefore, one might conceivably apply it in the context of gene networks.

4.1. *Classification-Based Approach*

In this subsection, we describe the classification-based method of Soinov *et al.*[783] for inferring molecular networks. Let a collection of n microarray gene expression output be given. For convenience, this collection can be organized into a gene expression matrix X. Each row of the matrix is a gene, each column is a sample, and each element $X[i,j]$ is the expression of gene i in sample j. Then the basic idea of the method of Soinov *et al.*[783] is as follows.

First determine the average value a_i of each gene i as $(\sum_j X[i,j])/n$. Next, denote s_{ij} as the state of gene i in sample j, where $s_{ij} = up$ if $X[i,j] \geq a_i$, and $s_{ij} = down$ if $X[i,j] < a_i$. Then, according to Soinov *et al.*,[783] to see whether the state of a gene g is determined by the state of other genes G, we check whether $\langle s_{ij} |\ i \in G \rangle$ can predict s_{gj}. If it can predict s_{gj} with high accuracy, then we conclude that the state of the gene g is determined by the states of other genes G.

Any classifier can be used to see if such predictions can be made reliably, such as C4.5,[693] PCL,[497] SVM,[855] and other classifiers described in Chapter 3.

Naturally, we can also apply feature selection methods described in Chapter 3—such as Fisher criterion score[251] or entropy-based methods[242]—to select a subset of genes from G before applying the classifiers to the selected subset of genes. Furthermore, to see how the state of a gene g is determined by the state of other genes, we apply C4.5, PCL, or other rule-based classifiers described in Chapter 3 to predict s_{gj} from $\langle s_{ij} |\ i \in G \rangle$ and extract the decision tree or rules used.

This interesting method has a few advantages: It can identify genes affecting a target genes in an explicit manner, it does not need a discretization threshold, each data sample is treated as an example, and explicit rules can be extracted from a rule-based classifier like C4.5 or PCL. For example, we generate from the gene expression matrix a set of n vectors $\langle s_{ij} |\ i \neq g \rangle \rightarrow s_{gj}$. Then C4.5 (or PCL) can be applied to see if $\langle s_{ij} |\ i \neq g \rangle$ predicts s_{gj}. The decision tree (or emerging patterns, respectively) induced would involve a small number of s_{ij}. Then we can suggest that those genes corresponding to these small number of s_{ij} affect gene g.

One other advantage of the Soinov method[783] is that it is generalizable to time series. Suppose the matrices X^t and X^{t+1} correspond to microarray gene expression measurements taken at time t and $t+1$. Suppose s_{ij}^t and s_{ij}^{t+1} correspond to the state of gene i in sample j at time t and $t+1$. Then to find out whether the state of a gene g is affected by other genes G in a time-lagged manner, we check whether $\langle s_{ij}^t |\ i \in G \rangle$ can predict s_{gj}^{t+1}. The rest of the procedure is as before.

Of course, there is a major caveat that this method as described assumes that a gene g can be in only two states, *viz.* $s_{gj} = up$ or $s_{gj} = down$. As cautioned by Soinov *et al.*,[783] it is possible for a gene to have more than two states and thus this assumption may not infer the complete network of gene interactions. Another caution is that if the states of two genes g and h are strongly co-related, the rules $s_{hj} \rightarrow s_{gj}$ and $s_{gj} \rightarrow s_{hj}$ saying that h depends on g and g depends on h are likely to be both inferred, even though only one of them may be true and the other false. Hence, further confirmation by experiments is advisable.

We do not have independent results on this approach to reconstructing molecular networks. However, we refer the curious reader to Soinov *et al.* [783] for a discussion on experiments they have performed to verify the relevance of this method. In particular, Soinov *et al.* have applied this method to the microarray datasets of Spellman and Cho for the *Saccharomyces cerevisiae* cell cycle.[157, 792] They consider a set of well-defined genes that encode proteins important for cell-cycle regulation and examine all extracted relations with respect to the known roles of the selected genes in the cell cycle. They have shown that in most cases the rules confirm the *a priori* knowledge.

4.2. *Association Rules Approach*

Recall from Chapter 3 that an association rule generally has the form $\alpha \rightarrow^{\mathcal{D}} \beta$, where α and β are disjoint sets of items, and the β set is likely to occur whenever the α set occurs in the context of a dataset $\mathcal{D}$. Note that we often drop the superscript $\mathcal{D}$ if the dataset $\mathcal{D}$ is understood or unimportant. As mentioned in Chapter 3, the support of an association rule $\alpha \rightarrow^{\mathcal{D}} \beta$ is the percentage of transactions in $\mathcal{D}$ that contains $\alpha \cup \beta$; and its confidence is the percentage of transactions in $\mathcal{D}$ containing α that also contain β.

In this subsection, we concentrate on the approach of Creighton and Hanash[176] for inferring associations between gene expression that is based on association rules. Let a collection of n microarray gene expression output be given as a gene expression matrix X so that each element $X[i, j]$ is the expression of gene i in sample j. Then the basic idea of the method of Creighton and Hanash is as follows.

Each element $X[i, j]$ is discretized into a state s_{ij} that indicates whether the gene i in sample j is considered up ($s_{ij} = up$), down ($s_{ij} = down$), or neither up nor down ($s_{ij} = neither$). This discretization to 3 states—up, $down$, and $neither$—is important because there is a good deal of noise in the data[373,823] and binning whole ranges of gene expression values into a few states is a good way to alleviate problems with noise. Creighton and Hanash[176] decide on the assignment of up, $down$, $neither$ by setting $s_{ij} = up$ if the expression value of gene i in sample j is greater than 0.2 for the log base 10 of the fold change, $s_{ij} = down$ if the expression value of gene i in sample j is less than –0.2 for the log base 10 of the fold change, and $s_{ij} = neither$ if the expression value of gene i in sample j is between –0.2 and 0.2 for the log base 10 of the fold change.

Then a dataset $\mathcal{D} = \{T_1, ..., T_n\}$ of n transactions is formed, where each sample j is treated as a transaction $T_j = \{gene_1 = s_{ij}, ..., gene_k = s_{kj}\}$. Then association rule mining algorithms described in Chapter 3 such as the Apriori algorithm[12] and the Max-Miner algorithm[70] can be used to mine for useful association rules. As many association rules can potentially be produced, Creighton and Hanash[176] adopt three measures for restricting the association rules to the most interesting ones, *viz.*

(1) they consider only those association rules that have support $\geq 10\%$ and confidence $\geq 80\%$;
(2) they consider only rules of the form $\alpha \rightarrow^{\mathcal{D}} \beta$ where α is a singleton; and
(3) they consider only the so-called "closed" rules, where a rule $\alpha \rightarrow^{\mathcal{D}} \beta$ is closed in the sense that there is no other rule $\alpha \rightarrow^{\mathcal{D}} \beta'$ such that $\beta \subset \beta'$ and has support $\geq 10\%$ and confidence $\geq 80\%$.

Creighton and Hanash[176] have applied this method to mine association rules from the gene expression profiles of 6316 transcripts corresponding to 300 diverse mutations and chemical treatment in yeast produced by Hughes *et al.*[373] They have obtained about 40 rules that contain ≥ 7 genes such as $\{YHM1 = up\} \rightarrow \{ARG1 = up, ARG4 = up, ARO3 = up, CTF13 = up, HIS5 = up, LYS1 = up, RIB5 = up, SNO1 = up, SNZ1 = up, YHR029C = up, YOL118C = up\}$. To see that these rules are significant, Creighton and Hanash also construct a randomized dataset and carry out association rule mining on this randomized dataset. On the randomized dataset, Creighton and Hanash is able to find only one rule. Hence, it is very likely that all the rules that are found by Creighton and Hanash from the dataset of Hughes *et al.* are not likely to have existed by chance.

This interesting method has two advantages. First, while we have made each transaction T_j to take the form $\{gene_1 = s_{1l}, ..., gene_k = s_{kj}\}$, it is possible to generalize it to include additional information such as environment and effects. As an example, consider $T_j = \{heatshock = 1, gene_1 = s_{1j}, ..., gene_k = s_{kj}\}$, where we use the item $heatshock = 1$ to indicate that a heat shock treatment has been first given to a sample j before profiling, and $heatshock = -1$ otherwise. Then we would be able to mine rules such as $\{heatshock = 1\} \rightarrow \{gene_h = up, gene_i = down\}$. That is, association rules may be helpful in relating the expression of genes to their cellular environment.

Second, the same gene is allowed to appear in several rules, in contrast to the clustering situation where each gene is normally required to appear in one cluster. A typical gene can participate in more than one gene network. Therefore, the association rule approach may be more useful in helping to uncover gene networks than the clustering approach. Furthermore, association rules also describe how the expression of one gene may be associated with the expression of a set of other genes.

Of course, there is a similar major caveat to that of the Soinov method.[783] This method as described above also assumes that a gene g can be in only three states, *viz.* $s_{gj} = up$, $s_{gj} = down$, or $s_{gj} = neither$. As cautioned by Soinov *et al.*,[783] it is possible for a gene to have more than three states and thus this assumption may not infer the complete network of gene interactions.

4.3. *Interaction Generality Approach*

In the two previous subsections, we have presented two techniques for inferring gene networks from microarray data. Both of these techniques can be said to work from a "positive" perspective in the sense that they assume there are no relation-

ship between the genes by default and attempt to directly infer rules that connect the state of one or more genes to the state of another gene.

Is it possible to work from a "negative" perspective in the sense of assuming every pair of genes affect each other by default and attempt to eliminate those that have no effect on each other? It turns out that this approach has been used in the related problem of eliminating false positive interactions from certain type of high-throughput protein-protein interaction experiments by Saito *et al.* [734,735]

A network of protein-protein interactions can be represented as an undirected graph $\mathcal{G}$, where each node represents a protein and each edge connecting two nodes represent an interaction between the two proteins corresponding to the two nodes. Given an edge $X \leftrightarrow Y$ connecting two proteins, X and Y, the "interaction generality" measure $ig^{\mathcal{G}}(X \leftrightarrow Y)$ of this edge as defined by Saito *et al.*[734] is equivalent to

$$ig^{\mathcal{G}}(X \leftrightarrow Y) = 1 + |\{X' \leftrightarrow Y' \in \mathcal{G} \mid X' \in \{X, Y\},\ deg^{\mathcal{G}}(Y') \geq 1\}|$$

where $deg^{\mathcal{G}}(U) = |\{V \mid U \leftrightarrow V \in \mathcal{G}\}|$ is the degree of the node U in the undirected graph $\mathcal{G}$. Note that in an undirected graph, an edge $X \leftrightarrow Y$ is the same one as the edge $Y \leftrightarrow X$. This measure is based on the idea that interacting proteins that appear to have many other interacting partners that have no further interactions are likely to be false positives.

Uetz *et al.*[849] and Ito *et al.*[386] independently screen yeast protein-protein interactions. Saito *et al.*[734] determine the interaction generality of all the interactions detected by the screens of Uetz *et al.* and Ito *et al.*. While only 72.8% of interactions that are detected exclusively by the screen of Ito *et al.* have interaction generalities ranging from 1 to 5, as many as 94.7% of interactions that are detected by both screens have interaction generalities ranging from 1 to 5. As the portion of protein-protein interactions that are detected in both screens are considered to be reliable—whereas those that are detected in one screen are considered very likely to be false positive interactions—this indicates that true positive interactions tend to be associated with low interaction generalities.

It is also widely accepted that interacting proteins are likely to share a common cellular role,[633] to be co-localized,[759] or to have similar gene expression profiles.[280,301,390] If interaction generality is indeed inversely related to true positive protein-protein interactions, then the proportion of protein-protein interaction pairs that share a common cellular role, that are co-localized or have similar gene expression profiles, must be increasing as we look at protein-protein interaction pairs of decreasing interaction generality. This is confirmed by Saito *et al.*[734] in the datasets of Ito *et al.* and Uetz *et al.*

The interaction generality measure of Saito *et al.*[734] does not take into consid-

eration the local topological properties of the interaction network surrounding the candidate interacting pair. Saito *et al.*[735] have also developed an improved interaction generality measure $ig_2^{\mathcal{G}}(X \leftrightarrow Y)$ that incorporates the local topological properties of interactions beyond the candidate interacting pair. They consider 5 local topological relationships between the candidate interacting pair and a third protein. The improved interaction generality measure is then computed as a weighted sum of the 5 topological relationships with respect to the third protein.

Most recently, our colleagues—Jin Chen, Wynne Hsu, Mong Li Lee, and See-Kiong Ng (private communication)—have proposed an "interaction pathway believeability" measure $ipb^{\mathcal{G}}(X \leftrightarrow Y)$ for assessing the reliability of protein-protein interactions obtained in large-scale biological experiments. It is defined as

$$ipb^{\mathcal{G}}(X \leftrightarrow Y) = \max_{\phi \in \Phi^{\mathcal{G}}(X,Y)} \prod_{(U \leftrightarrow V) \in \phi} \left(1 - \frac{ig^{\mathcal{G}}(U \leftrightarrow V)}{ig^{\mathcal{G}}_{\max}}\right)$$

where $ig^{\mathcal{G}}_{\max} = \max\{ig^{\mathcal{G}}(X \leftrightarrow Y) \mid (X \leftrightarrow Y) \in \mathcal{G}\}$ is the maximum interaction generality value in $\mathcal{G}$; and $\Phi^{\mathcal{G}}(X, Y)$ is the set of all possible non-reducible paths between X and Y, but excluding the direct path $X \leftrightarrow Y$. This measure can be seen as a measure on the global topological properties of the network involving X and Y in the sense that it evaluates the "credibility" of the non-reducible alternative path connecting X and Y, where the "probability" of each edge $U \leftrightarrow V$ in that path is $1 - ig^{\mathcal{G}}(U \leftrightarrow V)/ig^{\mathcal{G}}_{\max}$. Here, a path ϕ connecting X and Y is non-reducible if there is no shorter path ϕ' connecting X and Y that shares some common intermediate nodes with the path ϕ.

Jin Chen, Wynne Hsu, Mong Li Lee, and See-Kiong Ng further show that $ipb^{\mathcal{G}}(X \leftrightarrow Y)$ is better at separating true positive interactions from false positive interactions than $ig^{\mathcal{G}}(X \leftrightarrow Y)$ and $ig_2^{\mathcal{G}}(X \leftrightarrow Y)$. *E.g.*, on a large dataset of protein-protein interactions—comprising that of Uetz *et al.*[849], Ito *et al.*[386], and Mewes *et al.*[564]—the difference between the average value of $ig^{\mathcal{G}}(X \leftrightarrow Y)$ and $ig_2^{\mathcal{G}}(X \leftrightarrow Y)$ on true positive and false positive interactions are 7.37% and 7.83% respectively; but that of $ipb^{\mathcal{G}}(X \leftrightarrow Y)$ is 29.96%.

As mentioned earlier, Saito *et al.*[735] have identified 5 local topological relationships, between a candidate pair of interacting proteins and a third protein, that are particularly useful in distinguishing true positive protein-protein interactions from false positive interactions. Actually, Milo *et al.*[567] have also studied similar kind of local topological relationships in complex networks, including gene networks. They call these topological relationships network motifs. In particular, they[567] have reported two such network motifs for gene regulation networks of *E. coli* and *S. cerevisiae*. However, they have not explored using these network motifs to distinguish true positive interactions in gene networks from false positives.

5. Derivation of Treatment Plan

In Section 2, we see that the entropy measure can be used to identify genes that are relevant to the diagnosis of disease states and subtypes. Let us now end this chapter with a provocative idea of Li and Wong[496, 498] of the possibility of a personalized "treatment plan" that converts tumor cells into normal cells by modulating the expression levels of a few genes.

Let us use the colon tumour dataset of Alon *et al.*[21] to demonstrate this highly speculative idea. This dataset consists of 22 normal tissues and 40 colon tumor tissues. We begin with finding out which intervals of the expression levels of a group of genes occur only in cancer tissues but not in the normal tissues and vice versa. Then we attempt an explanation of the results and suggest a plan for treating the disease.

We use the entropy measure[242] described in Chapter 3 to induce a partition of the expression range of each gene into suitable intervals. This method partitions a range of real values into a number of disjoint intervals such that the entropy of the partition is minimal. For the colon cancer dataset, of its 2000 genes, only 135 genes can be partitioned into 2 intervals of low entropy. [496, 498] The remaining 1865 genes are ignored by the method. Thus most of the genes are viewed as irrelevant by the method.

For the purpose of this chapter we further concentrate on the 35 genes with the lowest entropy measure amongst the 135 genes. These 35 genes are shown in Figure 8. This gives us an easy platform where a small number of good diagnostic indicators are concentrated. For simplicity of reference, the index numbers in the first column of Figure 8 are used to refer to the two expression intervals of the corresponding genes. For example, the index 1 means M26338 $<$ 59.83 and the index 2 means M26383 $\geq$ 59.83.

An emerging pattern, as explained in Chapter 3, is a pattern that occurs frequently in one class of samples but never in other classes of samples. An efficient border-based algorithm[207, 495] is used to discover emerging patterns based on the selected 35 genes and the partitioning of their expression intervals induced by the entropy measure. Thus, the emerging patterns here are combinations of intervals of gene expression levels of these relevant genes.

A total of 10548 emerging patterns are found, 9540 emerging patterns for the normal class and 1008 emerging patterns for the tumour class. The top several tens of the normal class emerging patterns contain about 8 genes each and can reach a frequency of 77.27%, while many tumour class emerging patterns can reach a frequency of around 65%.

These top emerging patterns are presented in Figure 9 and Figure 10. Note

Our list	accession number	cutting points	Name
1,2	M26383	59.83	monocyte-derived neutrophil-activating ...
3,4	M63391	1696.22	Human desmin gene
5,6	R87126	379.38	myosin heavy chain, nonmuscle (Gallus gallus)
7,8	M76378	842.30	Human cysteine-rich protein (CRP) gene ...
9,10	H08393	84.87	COLLAGEN ALPHA 2(XI) CHAIN ...
11,12	X12671	229.99	heterogeneous nuclear ribonucleoprotein core ...
13,14	R36977	274.96	P03001 TRANSCRIPTION FACTOR IIIA
15,16	J02854	735.80	Myosin regulatory light chain 2 ...
17,18	M22382	447.04	Mitochondrial matrix protein P1 ...
19,20	J05032	88.90	Human aspartyl-tRNA synthetase alpha-2 ...
21,22	M76378	1048.37	Human cysteine-rich protein (CRP) gene ...
23,24	M76378	1136.74	Human cysteine-rich protein (CRP) gene ...
25,26	M16937	390.44	Human homeo box c1 protein mRNA
27,28	H40095	400.03	Macrophage migration inhibitory factor
29,30	U30825	288.99	Human splicing factor SRp30c mRNA
31,32	H43887	334.01	Complement Factor D Precursor
33,34	H51015	84.19	Proto-oncogene DBL Precursor
35,36	X57206	417.30	1D-myo-inositol-trisphosphate 3-kinase B ...
37,38	R10066	494.17	PROHIBITIN (Homo sapiens)
39,40	T96873	75.42	Hypothetical protein in TRPE 3'region ...
41,42	T57619	2597.85	40S ribosomal protein S6 ...
43,44	R84411	735.57	Small nuclear ribonucleoprotein assoc. ...
45,46	U21090	232.74	Human DNA polymerase delta small subunit
47,48	U32519	87.58	Human GAP SH3 binding protein mRNA
49,50	T71025	1695.98	Human (HUMAN)
51,52	T92451	845.7	Tropomyosin, fibroblast and epithelial ...
53,54	U09564	120.38	Human serine kinase mRNA
55,56	H40560	913.77	THIOREDOXIN (HUMAN)
57,58	T47377	629.44	S-100P PROTEIN (HUMAN)
59,60	X53586	121.91	Human mRNA for integrin alpha 6
61,62	U25138	186.19	Human MaxiK potassium channel beta subunit
63,64	T60155	1798.65	Actin, aortic smooth muscle (human)
65,66	H55758	1453.15	ALPHA ENOLASE (HUMAN)
67,68	Z50753	196.12	H.sapiens mRNA for GCAP-II/uroguanylin ...
69,70	U09587	486.17	Human glycyl-tRNA synthetase mRNA

Fig. 8. The 35 top-ranked genes by the entropy measure. The index numbers in the first column are used to refer to the two expression intervals of the corresponding genes. For example, the index 1 means M26338 $<$ 59.83 and the index 2 means M26383 $\geq$ 59.83.

Emerging patterns	Count & Freq. (%) in normal tissues	Count & Freq. (%) in cancer tissues
$\{25, 33, 37, 41, 43, 57, 59, 69\}$	17(77.27%)	0
$\{25, 33, 37, 41, 43, 47, 57, 69\}$	17(77.27%)	0
$\{29, 33, 35, 37, 41, 43, 57, 69\}$	17(77.27%)	0
$\{29, 33, 37, 41, 43, 47, 57, 69\}$	17(77.27%)	0
$\{29, 33, 37, 41, 43, 57, 59, 69\}$	17(77.27%)	0
$\{25, 33, 35, 37, 41, 43, 57, 69\}$	17(77.27%)	0
$\{33, 35, 37, 41, 43, 57, 65, 69\}$	17(77.27%)	0
$\{33, 37, 41, 43, 47, 57, 65, 69\}$	17(77.27%)	0
$\{33, 37, 41, 43, 57, 59, 65, 69\}$	17(77.27%)	0
$\{33, 35, 37, 41, 43, 45, 57, 69\}$	17(77.27%)	0
$\{33, 37, 41, 43, 45, 47, 57, 69\}$	17(77.27%)	0
$\{33, 37, 41, 43, 45, 57, 59, 69\}$	17(77.27%)	0
$\{13, 33, 35, 37, 43, 57, 69\}$	17(77.27%)	0
$\{13, 33, 37, 43, 47, 57, 69\}$	17(77.27%)	0
$\{13, 33, 37, 43, 57, 59, 69\}$	17(77.27%)	0
$\{13, 32, 37, 57, 69\}$	17(77.27%)	0
$\{33, 35, 37, 57, 68\}$	17(77.27%)	0
$\{33, 37, 47, 57, 68\}$	17(77.27%)	0
$\{33, 37, 57, 59, 68\}$	17(77.27%)	0
$\{32, 37, 41, 57, 69\}$	17(77.27%)	0

Fig. 9. The top 20 emerging patterns, in descending frequency order, in the 22 normal tissues. The numbers in the emerging patterns above refer to the index numbers in Figure 8.

that the numbers in the emerging patterns in these figures, such as $\{2, 10\}$ in Figure 10, refer to the index numbers in Figure 8. Hence, $\{2, 10\}$ denotes the pattern $\{\text{M26383} \geq 59.83, \text{H08393} \geq 84.87\}$.

The emerging patterns that are discovered are the most general ones. They occur in one class of data but do not occur in the other class. The discovered emerging patterns always contain only a small number of the relevant genes. This result reveals interesting conditions on the expression of these genes that differentiate between two classes of data.

Each emerging pattern with high frequency is considered as a common prop-

Emerging patterns	Count & Freq. (%) in normal tissues	Count & Freq. (%) in cancer tissues
$\{2, 10\}$	0	28 (70.00%)
$\{10, 61\}$	0	27 (67.50%)
$\{10, 20\}$	0	27 (67.50%)
$\{3, 10\}$	0	27 (67.50%)
$\{10, 21\}$	0	27 (67.50%)
$\{10, 23\}$	0	27 (67.50%)
$\{7, 40, 56\}$	0	26 (65.00%)
$\{2, 56\}$	0	26 (65.00%)
$\{12, 56\}$	0	26 (65.00%)
$\{10, 63\}$	0	26 (65.00%)
$\{3, 58\}$	0	26 (65.00%)
$\{7, 58\}$	0	26 (65.00%)
$\{15, 58\}$	0	26 (65.00%)
$\{23, 58\}$	0	26 (65.00%)
$\{58, 61\}$	0	26 (65.00%)
$\{2, 58\}$	0	26 (65.00%)
$\{20, 56\}$	0	26 (65.00%)
$\{21, 58\}$	0	26 (65.00%)
$\{15, 40, 56\}$	0	25 (62.50%)
$\{21, 40, 56\}$	0	25 (62.50%)

Fig. 10. The top 20 emerging patterns, in descending frequency order, in the 40 cancer tissues. The numbers in the emerging patterns refer to the index numbers in Figure 8.

erty of a class of cells. Based on this idea, Li and Wong[496, 498] propose a strategy for treating colon tumors by adjusting the expression level of some improperly expressed genes. That is, to increase or decrease the expression levels of some particular genes in a cancer cell, so that it has the common properties of normal cells and no properties of cancer cells. As a result, instead of killing the cancer cell, it is "converted" into a normal one. We show later that almost all "adjusted" cells are predicted as normal cells by a number of good classifiers that are trained to distinguish normal from colon tumor cells.

As shown in Figure 9, the frequency of emerging patterns can reach a very

high level such as 77.27%. The conditions implied by a highly frequent emerging pattern form a common property of one class of cells. Using the emerging pattern $\{25, 33, 37, 41, 43, 57, 59, 69\}$ from Figure 9, we see that each of the 77.27% of the normal cells simultaneously expresses the eight genes—M16937, H51015, R10066, T57619, R84411, T47377, X53586, and U09587 referenced in this emerging pattern—in such a way that each of the eight expression levels is contained in the corresponding interval—the 25th, 33th, 37th, 41st, 43rd, 57th, 59th, and 69th—as indexed in Figure 8.

Although a cancer cell may express some of the eight genes in a similar manner as normal cells do, according to the dataset, a cancer cell can never express all of the eight genes in the same way as normal cells do. So, if the expression levels of those improperly expressed genes can be adjusted, then the cancer cell can be made to have one more common property that normal cells exhibit. Conversely, a cancer cell may exhibit an emerging pattern that is a common property of a large percentage of cancer cells and is not exhibited in any of the normal cells. Adjustments should also be made to some genes involved in this pattern so that the cancer cell can be made to have one less common property that cancer cells exhibit. A cancer cell can then be iteratively converted into a normal one as described above.

As there usually exist some genes of a cancer cell which express in a similar way as their counterparts in normal cells, less than 35 genes' expression levels are required to be changed. The most important issue is to determine which genes need an adjustment. The emerging patterns can be used to address this issue as follows. Given a cancer cell, first determine which top emerging pattern of normal cells has the closest Hamming distance to it in the sense that the least number of genes need to be adjusted to make this emerging pattern appear in the adjusted cancer cell. Then proceed to adjust these genes. This process is repeated several times until the adjusted cancer cell exhibits as many common properties of normal cells as a normal cell does. The next step is to look at which top emerging pattern of cancer cells that is still present in the adjusted cancer cell has the closest Hamming distance to a pattern in a normal cell. Then we also proceed to adjust some genes involved in this emerging pattern so that this emerging pattern would vanish from the adjusted cancer cell. This process is repeated until all top emerging patterns of cancer cells disappear from our adjusted cancer cell.

We use a cancer cell (T1) of the colon tumor dataset as an example to show how a tumor cell is converted into a normal one. Recall the emerging pattern $\{25, 33, 37, 41, 43, 57, 59, 69\}$ is a common property of normal cells. The eight genes involved in this emerging pattern are M16937, H51015, R10066, T57619, R84411, T47377, X53586, and U09587. Let us list the expression profile of these

eight genes in T1:

genes	expression levels in T1
M16937	369.92
H51015	137.39
R10066	354.97
T57619	1926.39
R84411	798.28
T47377	662.06
X53586	136.09
U09587	672.20

However, 77.27%—17 out of 22 cases—of the normal cells have the following expression intervals for these 8 genes:

genes	expression interval
M16937	$<$390.44
H51015	$<$84.19
R10066	$<$494.17
T57619	$<$2597.85
R84411	$<$735.57
T47377	$<$629.44
X53586	$<$121.91
U09587	$<$486.17

Comparing T1's gene expression levels with the intervals of normal cells, we see that 5 of the 8 genes—H51015, R84411, T47377, X53586, and U09587—of the cancer cell T1 behave in a different way from those the 22 normal cells commonly express. However, the remaining 3 genes of T1 are in the same expression range as most of the normal cells. So, if the 5 genes of T1 can be down regulated to scale below those cutting points, then this adjusted cancer cell will have a common property of normal cells. This is because $\{25, 33, 37, 41, 43, 57, 59, 69\}$ is an emerging pattern which does not occur in the cancer cells. This idea is at the core of Li and Wong[496, 498]'s suggestion for this treatment plan.

Interestingly, the expression change of the 5 genes in T1 leads to a chain of

other changes. These include the change that 9 extra top-ten EPs of normal cells are contained in the adjusted T1. So all top-ten EPs of normal cells are contained in T1 if the 5 genes' expression levels are adjusted. As the average number of top-ten EPs contained in normal cells is 7, the changed T1 cell will now be considered as a cell that has the most important features of normal cells. Note that we have adjusted only 5 genes' expression level so far.

It is also necessary to eliminate those common properties of cancer cells that are contained in T1. By adjusting the expression level of 2 other genes, M26383 and H08393, the top-ten EPs of cancer cells all disappear from T1. According to the colon tumor dataset, the average number of top-ten EPs of cancer cells contained in a cancer cell is 6. Therefore, T1 is converted into a normal cell as it now holds the common properties of normal cells and does not hold the common properties of cancer cells.

By this method, all the other 39 cancer cells can be converted into normal ones after adjusting the expression levels of 10 genes or so, possibly different genes from person to person. Li and Wong[496, 498] conjecture that this personalized treatment plan is effective if the expression of some particular genes can be modulated by suitable means.

Lastly, we discuss a validation of this idea. The "adjustments" made to the 40 colon tumour cells are based on the emerging patterns in the manner described above. If these adjustments have indeed converted the colon tumour cells into normal cells, then any good classifier that can distinguish normal vs colon tumour cells on the basis of gene expression profiles is going to classify our adjusted cells as normal cells. So, Li and Wong[496, 498] establish a SVM model using the original entire 22 normal plus 40 cancer cells as training data. The code for constructing this SVM model is available at `http://www.cs.waikato.ac.nz/ml/weka`. The prediction result is that all of the adjusted cells are predicted as normal cells. Although Li and Wong's "therapy" is not applied to the real treatment of a patient, the prediction result by the SVM model partially demonstrates the potential biological significance of this highly speculative and provocative proposal.