Cluster Analysis and its Applications to Gene Expression Data

R. SHARAN[1]

R. ELKON[2]

R. SHAMIR[1]

[1]School of Computer Science, Tel-Aviv University, Tel-Aviv 69978, Israel.

{roded,rshamir}@post.tau.ac.il.

[2] The David and Inez laboratory for genetic research, Department of Human Genetics and Molecular Medicine, Sackler School of Medicine, Tel-Aviv University, Tel-Aviv 69978, Israel.

ranel@post.tau.ac.il.

1. Introduction

Technologies for generating high-density arrays of cDNAs and oligonucleotides are developing rapidly, and changing the landscape of biological and biomedical research. They enable, for the first time, a global, simultaneous view on the transcription levels of many thousands of genes, when the cell undergoes specific processes and in certain conditions. For several organisms, the sequences of all genes are available, and thus, transcript levels of the complete gene collection can already be monitored today. The potential of such technologies is tremendous: Monitoring gene expression levels in different developmental stages, tissue types, clinical conditions and different organisms can help understanding gene function and gene networks, assist in the diagnosis of disease conditions and reveal the effects of medical treatments. Undoubtedly, other applications will emerge in coming years.

A word on terminology: *gene expression data* is usually represented by a matrix, with rows corresponding to genes, and columns corresponding to conditions, experiments or time points. The content of the matrix is the expression levels of each gene under each condition. Those levels may be absolute, relative or otherwise normalized. Each column contains the results obtained from a single array in a particular condition, and is called the *profile* of that condition. Each row vector is the *expression pattern* of a particular gene across all the conditions. More formal definitions will be given in the sequel.

What is Clustering: A key initial step in the analysis of gene expression data is the detection of groups of genes that exhibit similar expression patterns. This translates to the algorithmic problem of clustering. A clustering problem consists of elements and a characteristic vector for each element. A measure of similarity is defined between pairs of such vectors. (In gene expression, elements are usually genes and the vector of each gene is its expression pattern; similarity can be measured in various ways that are problem dependent, for example, by the correlation coefficient between vectors.) The goal is to partition the elements into subsets, which are called *clusters*, so that two criteria are satisfied: *Homogeneity* - elements in the same cluster are highly similar to each other; and *separation* - elements from different clusters have low similarity to each other.

The Array Technology: There are currently two main technologies that generate large-scale gene expression data. Both are based on performing a large number of hybridizations in parallel in a single experiment on high-density arrays (a.k.a. "DNA chips"), between probes and targets. *cDNA microarrays* (Schena et al. 1996, Schena 1996, Marshall and Hodgson 1998, Ramsay 1998) contain large sets of cDNA sequences, each several hundred bases long, immobilized on a solid substrate. In an array experiment, many gene-specific cDNAs are spotted on a single matrix. The matrix is then simultaneously probed with cDNA representations of total RNA pools from test and reference cells, tagged with distinct fluorescent dyes, allowing one to determine the relative amount of transcript present in the pool by the relative intensities of the fluorescent signals generated at each spot. In *oligonucleotide microarrays* (Fodor et al. 1993, Lipshutz et al. 2000, Harrigton et al. 2000), each gene is represented on the array by a set of 15-20 oligonucleotide probes, typically 25 bases long, designed to hybridize to different regions of the same RNA. This use of multiple detectors greatly improves reproducibility and accuracy of RNA quantification, and reduces the rate of false-positives and miscalls.

Outline of this Chapter: In this chapter we shall describe applications of clustering to gene expression data. We shall outline some of the popular algorithms in this field, and focus mainly on the CLICK algorithm developed by Sharan and Shamir (2000). In clustering of genes, we shall present results of CLICK in comparison to the other algorithms. Some of these results were previously reported in (Sharan and Shamir 2000,2001). The latter also contains a broader exposition of other clustering methods. In addition, we shall present results of using CLICK to find common regulatory motifs in upstream regions of clustered genes, and to classify tissues based on their expression profiles. The results demonstrate the utility of clustering in general, and CLICK in particular, in a wide variety of applications to gene expression analysis.

2. Algorithms

In this section we first give basic mathematical background on clustering. We then describe three clustering algorithms used for gene expression analysis. We end the section with a discussion on how to measure the quality of a suggested clustering solution.

2.1 Mathematical Background

Let $N=\{e_1,\ldots,e_n\}$ be a set of n elements, and let $C=\{C_1,\ldots,C_l\}$ be a *partition* of N into disjoint subsets. Each subset is called a *cluster*, and C is called a *clustering solution*, or simply a *clustering*. Two elements e_i and e_j are called *mates with respect to C* if they are members of the same cluster in C. In the gene expression context, the elements are the genes and we often assume that there exists some correct partition of the genes into “true” clusters. Alternatively, the elements are the conditions or tissues, that are assumed to belong to one of several categories, e.g., tumor or normal tissues. When C is the true clustering of N, elements that belong to the same true cluster are simply called *mates*.

The input data for a clustering problem is typically given in one of two forms: (I) *Fingerprint data* - each element is associated with a real-valued vector, called its *fingerprint*, or *pattern*, which contains p measurements on the element, e.g., expression levels of an mRNA at p different conditions (cf. Eisen and Brown 1999). (II) *Similarity data* - pairwise similarity values between elements. These values can be computed from fingerprint data, e.g., by correlation between vectors. Alternatively, the data can represent pairwise dissimilarity, e.g., by computing distances. Fingerprints contain more information than similarity values, but the latter are completely generic and can be used to represent the input to clustering in any application. (Note that there is also a practical storage consideration regarding the presentation: The fingerprint matrix is of order $n \times p$ while the similarity matrix is of order $n \times n$. In gene expression applications often $n >> p$. In contrast, in tissue classification applications often $n << p$.)

The goal in a clustering problem is to partition the set of elements into homogeneous and well-separated clusters. That is, we require that elements from the same cluster will be highly similar to each other, while elements from different clusters will have low similarity to each other. This “meta-formulation” does not define a specific optimization problem, since homogeneity and separation can be mathematically formulated in various ways, leading to a variety of optimization problems. Note also, that even when homogeneity and separation are precisely defined, those are two opposing objectives: The better the homogeneity – the poorer the separation, and vice versa.

Clustering problems and algorithms are often represented in graph-theoretic terms and we shell use this representation here. We therefore include some basic definitions on graphs. We refer the readers to Golumbic (1980) and Even (1979) for more background and terminology on graphs.

Let *G=(V,E)* be a weighted graph. We denote the vertex set *V* of *G* also by *V(G)*. For a subset *R* of *V*, the *subgraph induced by R*, is obtained from *G* by deleting all vertices not in *R* and the edges incident on them. The *weight* of a vertex *v* is the sum of the weights of the edges incident on *v*. A *cut C* in *G* is a subset of its edges, whose removal disconnects *G*. The *weight* of *C* is the sum of the weights of its edges. A *minimum weight cut* is a cut in *G* with minimum weight.

For a set of elements $K \subseteq N$, we define the *fingerprint* or *centroid* of *K* to be the mean vector of the fingerprints of the members of *K*. For two fingerprints *x* and *y* we denote their similarity by *S(x,y)* and their dissimilarity by *d(x,y).* A *similarity graph* is a weighted graph in which vertices correspond to elements and edge weights are derived from the similarity values between the corresponding elements. Hence, the similarity graph is just an equivalent representation of the similarity matrix.

An alternative formulation of the clustering problem is hierarchical: Rather than asking for a single partition of the elements, one seeks an iterated partition: A *dendogram* is a rooted weighted tree, with leaves corresponding to elements. The customary (perhaps counter-intuitive) way is to present the tree with the root at the top and the leaves at the bottom. Each edge defines the cluster of elements contained in the subtree below that edge. The edge's weight (or length) reflects the dissimilarity between that cluster and the remaining elements. In this formulation the clustering solution is the dendogram, and each non-singleton cluster, corresponding to a rooted subtree, is split into subclusters. The determination of disjoint clusters is left to the judgment of the user. Typically, one tends to consider as a genuine cluster the set of elements of a subtree just below a connecting edge of high weight.

Irrespective of the representation of the clustering problem input, judicious preprocessing of the raw data is essential to meaningful clustering. This preprocessing is application dependent and must be chosen in view of the expression technology used and the biological questions asked. The goal of the preprocessing is to normalize the data and calculate the pairwise element (dis)similarity, if applicable. Common procedures for normalizing fingerprint data include transforming each fingerprint to have mean zero and variance one, a fixed norm or a fixed maximum entry. Statistically based methods for data normalization have also been developed recently (cf. Kerr et al. 2000).

Several algorithmic techniques were previously used for clustering gene expression data, including hierarchical clustering (Eisen et al. 1998), self organizing maps (Tamayo et al. 1999), and graph theoretic approaches (Sharan and Shamir 2000). We describe these approaches in the sequel. For other approaches to clustering expression patterns, see (Ben-Dor et al. 1999, Hartuv and Shamir 2000, Milosavljevic et al. 1995, Alon et al. 1999, Getz et al. 2000, Heyer et al. 1999). For more extensive reviews and more information and background on clustering, see (Hartigan 1975, Everitt 1993, Mirkin 1996, Hansen and Jaumard 1997, Shamir and Sharan 2001).

2.2 Hierarchical Clustering

Hierarchical clustering solutions are typically represented by a dendogram. Algorithms for generating such solutions often work either in a top-down manner, by repeatedly partitioning the set of elements, or in a bottom-up fashion. We shall describe here the latter approach. Such *agglomerative hierarchical clustering* algorithms are among the oldest and most popular clustering methods (Cormack 1971). They proceed from an initial partition into singleton clusters by successive merging of clusters until all elements belong to the same cluster. Each merging step corresponds to joining two clusters. The general scheme due to Lance and Williams (1967) follows. It is assumed that $D=(d(i,j))$ is the input dissimilarity matrix.

The agglomerative hierarchical clustering scheme:

1. Find a minimal entry $d(i,j)$ in D, and merge clusters i and j.
2. Modify D by deleting rows and columns i,j and adding a new row and column $i\cup j$, with their dissimilarities defined by: $d(k,i\cup j)=d(i\cup j,k)=\alpha_i d(k,i)+\alpha_j d(k,j)+\gamma|d(k,i)-d(k,j)|$
3. If there is more than one cluster, then go to Step 1.

Common variants of this scheme, obtained for appropriate choices of the α-s and γ parameters, are the following:

Single-linkage: $d(k, i\cup j)=min\{d(k,i), d(k,j)\}$.

Complete-linkage: $d(k, i\cup j)=max\{d(k,i), d(k,j)\}$.

Average-linkage: $d(k, i\cup j)= (n_i\cdot d(k,i) + n_j\cdot d(k,j))/(n_i+n_j)$, where n_i denotes the number of elements in cluster *i*.

Eisen et al. (1998) developed a clustering software package based on average-linkage hierarchical clustering. The clustering program is called Cluster, and the accompanying visualization program is called TreeView. Both programs are available at http://rana.stanford.edu/software/. The gene similarity metric used is a form of correlation coefficient. The algorithm iteratively merges elements whose similarity value is the highest, as explained above. The output of the algorithm is a dendogram and an ordered fingerprint matrix. The rows in the matrix are reordered based on the dendogram, so that groups of genes with similar expression patterns are adjacent.

2.3 Self Organizing Maps

The self organizing map (SOM) was developed by Kohonen (1997) as a method for fitting a number of ordered discrete reference vectors to the distribution of vectorial input samples. The method assumes that the number of clusters is known. Those clusters are organized as a set of nodes in a hypothetical "elastic network", with a simple neighborhood structure on the nodes, e.g., a two-dimensional $k\times l$ grid. Each node *n* in the grid is associated with a reference vector. In the process of running the algorithm, the input vectors direct the movement of the reference vectors towards the denser areas of the input vectors space, so that an organization of the input vectors over the network emerges. In the following we describe the SOM algorithm in the Euclidean space.

The SOM process is iterative. Denote by f_i(n) the position of the reference vector of node *n* at the *i*-th iteration. The initial positioning f_1 is random. The algorithm iteratively selects a random data point *p*, identifies the node n_p whose reference vector $f_i(n_p)$ is closest to p, and updates the position of all reference vectors towards p according to a predefined learning function $l(\cdot)$:

$$f_{i+1}(n) = f_i(n) + l(n, n_p, i)[p - f_i(n)]$$

The amount of position adjustment determined by $l(n,n_p,i)$ decreases as the distance between n and n_p (in the grid) and the iteration number grow. The intuition for this learning process is that the reference vectors that are close enough to p will be pulled towards it, and the stiffness of the grid structure will propagate some of impact to neighboring nodes. For much more on self organizing maps the reader is referred to Kohonen (1997).

Tamayo et al. (1999) devised a gene expression clustering software, GeneCluster, which uses the SOM algorithm. The software is available at http://waldo.wi.mit.edu/MPR/. In their implementation they incorporated a neighborhood learning function, which is zero for nodes distant from n_p and equals $0.02T/(T+100i)$ for nodes close to n_p, where T is the maximum number of iterations. GeneCluster accepts an input file of expression levels together with a two dimensional grid geometry for the nodes. The number of grid points is the prescribed number of clusters. The resulting clusters are visualized by presenting for each cluster its average expression pattern along with error bars displaying the standard deviation at each condition. Clusters are presented in their grid order, as clusters of close nodes tend to be similar.

Another implementation of SOM for clustering gene expression profiles was developed by Toronen et al. (1999).

2.4 CLICK

The CLICK (CLuster Identification via Connectivity Kernels) algorithm (Sharan and Shamir 2000) employs a graph theoretic approach to clustering. The software is available at http://www.math.tau.ac.il/~rshamir/click.html. The algorithm first preprocesses the input data and forms a weighted similarity graph. It then recursively partitions the current set of elements into two subsets. Before a partition, the algorithm tests if the subgraph induced by the current subset of elements is a *kernel* of a cluster (the definition of a kernel is given below). If this is the case, the subgraph is not partitioned further. Otherwise, a minimum weight cut is computed in the subgraph, and the current set of elements is split into the two subsets separated by that cut. The output is a list of kernels which serve as a basis for the eventual clusters, and a set of singletons, i.e., single vertices to be handled later. This scheme is detailed below. It is assumed that procedure MinWeightCut(G) computes a minimum weight cut of G and returns a partition of G into two subgraphs H and K according to this cut.

```
Form-Kernels($G$):
If $V(G)=\{v\}$ then move $v$ to the singleton set.
Else if $G$ is a kernel then output $V(G)$.
Else
    $(H,K) \leftarrow$ MinWeightCut($G$).
    Form-Kernels($H$).
    Form-Kernels($K$).
```

CLICK builds on a statistical model, which gives probabilistic meaning to edge weights in the similarity graph and to the stopping criterion. The key probabilistic assumption of CLICK is that pairwise similarity values between elements are normally distributed: Similarity values between mates are normally distributed with mean μ_T and variance σ_T^2, and similarity values between non-mates are normally distributed with mean μ_F and variance σ_F^2, where $\mu_T>\mu_F$. This situation often holds on real data, and can be asymptotically justified under certain assumptions (Sharan and Shamir 2000).

The algorithm uses the values of these distribution parameters as well as the probability p_{mates} that two randomly chosen elements are mates. These parameters can be computed directly from a known solution on a subset of the elements (such a solution is often available, e.g., in oligofingerprinting experiments (Poustka et al. 1999)). Alternatively, the parameters can be estimated using the EM algorithm, assuming the above probabilistic model (see, e.g., Mirkin 1996, Sec. 3.2.7).

Let $S=(S_{ij})$ be the input similarity matrix. Form a weighted similarity graph G in which the weight w_{ij} of the edge (i,j) reflects the probability that i and j are mates, and is derived from the normal density function and Bayes Theorem:

$$w_{ij} = \ln\frac{\Pr(i, j \text{ are mates} \mid S_{ij})}{\Pr(i, j \text{ are non-mates} \mid S_{ij})} = \ln\frac{p_{mates}\sigma_F}{(1-p_{mates})\sigma_T} + \frac{(S_{ij}-\mu_F)^2}{2\sigma_F{}^2} + \frac{(S_{ij}-\mu_T)^2}{2\sigma_T{}^2}$$

The current subgraph is determined to be a kernel if the value of a minimum cut in it is positive. This is the case if and only if for every cut C in the current subgraph, the probability that it contains only edges between mates exceeds the probability that C contains only edges between non-mates.

The actual implementation omits from the graph all edges with values below some predefined non-negative threshold, computes the minimum cut in that simplified graph using the Hao-Orlin algorithm (1994), and corrects the solution value for the missing edges. In order to reduce running time on very big instances, a screening heuristic removes low weight vertices from large components. The removed vertices are marked as singletons and handled at a later stage.

CLICK first produces kernels which form the basis of the eventual clusters. Subsequent processing includes *singleton adoption*, recursive clustering process on the set of remaining singletons, and an iterative *merging step*. The singletons adoption step is based on computing similarities between singletons' and clusters' fingerprints. The merging step iteratively merges two kernels whose fingerprint similarity is the highest, provided that this similarity exceeds a predefined threshold. Visualization of the clusters shows average expression patterns and standard error bars for each cluster.

2.5 Assessment of solutions

A key question in the design and analysis of clustering techniques is how to evaluate solutions. We present in this section figures of merit for measuring the quality of a clustering solution. Different measures are applicable in different situations, depending on whether a (partial) true solution is known or not, and whether the input is fingerprint or merely similarity data. We describe below some of the applicable measures in case the true solution is unknown. For other possible figures of merit we refer the reader to (Everitt 1993, Hansen and Jaumard 1997, Yeung et al. 2000).

When the true solution is not known, we evaluate the quality of a suggested solution by computing two figures of merit that measure its homogeneity and separation. For fingerprint data, homogeneity is evaluated by the average similarity between the fingerprint of an element and that of its cluster. Precisely, if *cl(u)* is the cluster of *u*, *F(X)* and *F(u)* are the fingerprints of a cluster *X* and an element *u*, respectively, and S is the similarity function, then

$$H_{Ave} = \frac{1}{N} \sum_{u \in N} S(F(u), F(cl(u)))$$

Separation is evaluated by the weighted average similarity between cluster fingerprints. That is, if the clusters are X$_1$,…,X$_t$, then

$$S_{Ave} = \frac{1}{\sum_{i \neq j} | X_i || X_j |} \sum_{i \neq j} | X_i || X_j | S(F(X_i), F(X_j))$$

Hence, a solution improves if H$_{Ave}$ increases, and if S$_{Ave}$ decreases. In computing the above measures singletons are considered as additional one-member clusters.

3. Clustering Genes

In this section we describe three applications of CLICK for forming gene clusters on previously analyzed datasets. For the first dataset we present a comparison with GeneCluster. On the second, a comparison with Cluster. For the third dataset we present a comparison among CLICK, GeneCluster, CAST (Ben-Dor et al., 1999) and K-Means (Herwig et al., 2000).

3.1 Yeast Cell-Cycle

The yeast cell cycle dataset of Cho et al. (1999) contains the expression levels of 6,218 *S. cerevisiae* putative gene transcripts (identified as ORFs) measured at 10-minutes intervals over two cell cycles (160 minutes). We compared CLICK's results to those of GeneCluster (Tamayo et al. 1999). To this end, we applied the same filtering and data normalization procedures of (Tamayo et al. 1999). The filtering removes genes which do not change significantly across samples, resulting in a set of 826 genes. Additionally, the 90-minutes time-point was removed. The expression levels of each gene were normalized to have mean zero and variance one within each of the two cell-cycles. CLICK clustered the genes into 30 clusters in 14 seconds, leaving 55 unclustered singletons. A summary of the homogeneity and separation parameters for the solutions produced by CLICK and GeneCluster is shown in Table 1. CLICK obtained results superior in both parameters.

Table1

Two putative true cluster are the sets of late G1-peaking genes and M-peaking genes, reported in (Cho et al. 1999). Out of the late G1-peaking genes that passed the filtering, CLICK placed 91% in a single cluster. In contrast, Tamayo et al. (1999) report that in their solution 87% of these genes were contained in three clusters. Out of the M-peaking genes that passed the filtering, CLICK placed 95% in a single cluster, while in the solution of (Tamayo et al. 1999) 92.5% of the genes are again partitioned among three different clusters.

3.2 Serum Response

The dataset of Iyer et al. (1999) contains the expression levels of 8,613 human genes obtained as follows: Human fibroblasts were deprived of serum for 48 hours and then stimulated by addition of serum. Expression levels of genes were measured at 12 time-points after the stimulation. An additional data-point was obtained from a separate unsynchronized sample. A subset of 517 genes whose expression levels changed substantially across samples was analyzed by the hierarchical clustering method of Eisen et al. (1998). The data was normalized by dividing each entry by the expression level at time zero, and taking a logarithm of the result. For ease of manipulation, we also transformed each fingerprint to have a fixed norm. The similarity function used was dot-product, giving values identical to those used by Eisen et al. (1998). CLICK clustered the genes into 10 clusters in 32 seconds, leaving 26 singletons. These clusters are shown in Figure 1. Table 2 presents a comparison between the clustering quality of CLICK and the hierarchical clustering of Eisen et al. (1998) on this dataset. (The determination of clusters from the dendogram was done manually by Eisen et al.) Again, CLICK performs better in both parameters.

Table 2

3.3 Yeast Cell-Cycle Revisited

Here we describe a 'blind' comparison among several clustering algorithms, including CLICK, CAST, GeneCluster and K-Means, on a yeast cell-cycle dataset containing the gene expression levels of yeast ORFs over 79 conditions. The dataset is available at http://cellcycle-www.stanford.edu. This comparison was originally reported in (Shamir and Sharan 2001). The original dataset (Spellman et al. 1998) contains samples from yeast cultures synchronized by four independent methods: α factor arrest (samples taken every 7 minutes for 119 minutes); arrest of a cdc15 temperature sensitive mutant (samples taken every 10 minutes for 290 minutes); arrest of a cdc28 temperature sensitive mutant (this part of the data is from (Cho et al. 1999), as described in section 3.1), and elutriation (samples taken every 30 minutes for 6.5 hours). It also contains separate experiments in which G1 cyclin Cln3p or B-type cyclin Clb2p were induced. Spellman et al. identified in this data 800 genes that are cell-cycle regulated (Spellman et al. 1998). The dataset that we used contains the expression levels of 698 out of those 800 genes, which have no missing entries, over the 72 conditions that cover the α factor, cdc28, cdc15, and elutriation experiments. As in (Tamayo et al. 1999), the 90 minutes datapoint was omitted from the cdc15 experiment. Each row of the 698×72 matrix was normalized to have mean 0 and variance 1. (Note that by normalizing the variance different gene amplitudes are deemphasized and periodicity is made more prominent).

Based on the analysis conducted by Spellman et al. we expect to find in the data five main clusters: G1-peaking genes, S-peaking genes, G2-peaking genes, M-peaking genes, and M/G1-peaking genes. Each of these clusters was shown to contain biologically meaningful sub-clusters. The dataset was clustered using four methods: K-Means (Herwig et al. 2000), GeneCluster (Tamayo et al. 1999), CAST (Ben-Dor et al. 1999), and CLICK (Sharan and Shamir 2000). The similarity measure used was Pearson correlation coefficient. The authors of each of the programs were given the dataset and asked to provide a clustering solution. The identity of the dataset was not described and genes were permuted in an attempt to disguise the data source. (Admittedly, the test was not really blind, since someone familiar with the gene expression literature could have identified the nature of the data.) The authors were told in advance that the criteria of average homogeneity and average separation would be used to evaluate the quality of the solutions.

Table 3 summarizes the solutions produced by each program, and their homogeneity and separation parameters. The so-called 'True' clustering, reported in Spellman et al. (1998) was obtained manually by comparing the expression patterns and the literature and is not very accurate (M. Eisen, private communication). The solution produced by CLICK contains 67 unclustered singletons.

Table 3

4. Identifying regulatory motifs

Global gene expression data enables the delineation of genetic regulatory networks via direct or indirect approaches. In the direct approach, the effects of activation or repression of a specific transcription factor (TF) on gene expression are monitored. This way, genes located downstream from p53 (Zhao et al. 2000), BRCA1 (Harkin et al. 1999) and C-myc (Coller et al. 2000) in their respective pathways, were identified.

The indirect approach for the delineation of regulatory networks relies on the hypothesis that genes exhibiting similar expression patterns across a large panel of biological conditions share common regulatory elements in their promoter regions. In other words, co-expression is correlated with co-regulation (Tavazoie et al. 1999, Zhang 1999, Brazma and Vilo 2000). The regulatory elements in the promoter region represent the "switches" that respond to signals from various cellular signaling pathways. The response can be either as part of the normal developmental program of the organism, or in response to external perturbations, stresses and alterations in physiological conditions. The binding of transcription factors to their binding sites in the promoter region enhances (or represses) the transcription initiation complex recruitment and assembly on the basal promoter in the proximity of the transcription start site, thereby influencing transcription initiation.

The approach that aims to detect cis-regulatory TF's binding sites from co-expression comprises two steps: (I) Cluster analysis aimed at the identification of clusters of genes sharing similar expression patterns. (II) Sequence analysis, which searches for sequence patterns that are over-represented in upstream regions of members of the same cluster. The derivation of regulatory networks through the identification of common cis-regulatory elements shared by co-regulated genes was successfully demonstrated in yeast (Spellman et al. 1998, Cho et al. 1998, Jelinsky et al. 2000) and arabidopsis thaliana (Maleck et al. 2000).

In order to test the utility of CLICK for motif identification, we have analyzed the dataset published recently by Jelinsky et al. (2000). In that experiment, gene expression levels of all 6,200 ORFs of the yeast Saccharomyces Cerevisiae were measured in order to study the cellular response to DNA damage. In total, gene expression profiles in 26 biological conditions were measured, including treatments with various DNA damaging agents at several time points and doses. 2,610 genes that changed by a factor of 3 or more in at least one condition were subjected to cluster analysis. The clustering reported in (Jelinsky et al. 2000) consists of 18 clusters, obtained by GeneCluster. In comparison, CLICK identified 33 clusters with more than 10 members.

Once the clusters are identified, the motif finding algorithm is applied to promoter regions of the genes in each cluster. In this study, the search was performed on the 500 bases upstream to the ORFs' translation start sites. The analysis was performed using the AlignACE package (http://atlas.med.harvard.edu/, Roth et al. 1998, Hughes et al. 2000) as done in (Jelinsky et al. 2000). (As the motif finding software was recently modified, in addition to its application to CLICK's clustering, we also reapplied AlignACE to the clustering reported in (Jelinsky et al. 2000).) AlignACE employs Gibbs sampling for detecting over-represented motifs in a target set of sequences. It utilizes the mononucleotide frequencies as genomic background. To focus on regulatory motifs that potentially form the mechanistic basis for the observed co-expression, as well as to suppress false-negatives, only motifs that exceed two score thresholds are reported. The first is an *alignment score*, which gauges the statistical significance of the identified motif over the genomic background. The second is a *specificity score*, which gauges how specific is the identified motif to promoters of genes in the cluster, relative to promoter regions of other genes in the genome (Hughes et al. 2000).

In total, 26 significant motifs were identified in CLICK's clusters, and 30 such motifs were identified in GeneCluster's clusters. The identified motifs were matched against the SCPD database of experimentally verified yeast's TF binding sites (http://cgsigma.cshl.org/jian/, Zhu and Zhang 1999). Table 4A summarizes the number of motifs identified in each clustering as well as the number of motifs that had a match in the SCPD DB. For both clustering methods, more than 60% of the motifs had a verified TF binding site match, a fact that indicates the utility of this approach. In addition, motifs with no match to known binding site were detected as well. Each of these motifs forms a hypothesis that should be subjected to further biological research.

Of the 26 motifs identified in CLICK's clusters, 17 were common with GeneCluster's motifs. Common motifs were identified using CompareAce (part of the AlignAce package), which calculates a similarity coefficient for pairs of input motifs. Motifs whose similarity exceeded a threshold of 0.7 were regarded as common. Table 4B lists those common motifs. It is interesting to note that the percent of verified common motifs is particularly high (more than 75%, see Table 4A). Hence, the four common, unidentified motifs are more likely to be true, as they were obtained by two different methods.

The delineation of cis-regulatory motifs via analysis of promoters of co-regulated genes was so far demonstrated in model organisms where the identification of promoter regions is straightforward. Most TF binding sites are located within a few hundred bases upstream from the transcription start site. As the 5'-untranslated regions (UTRs) are very short (100-200b), the predicted translation start sites are very proximate to the transcription start sites (most of which are unmapped). Therefore, sequence analysis aimed at detecting the regulatory elements, can be conducted on sequences in the upstream vicinity of the easy-to-determine translation start sites. In contrast, the identification of promoters in mammals is much more difficult, as the 5'-UTR can be very long (up to several kilobases). However, several full length cDNA projects are being conducted, which will substantially increase the number of genes for which transcription start sites are mapped (Werner 2001). A recently published research indeed demonstrated that this approach for deciphering regulatory mechanism can be successfully implemented in mammals, provided that promoters of co-regulated genes are mapped (Livesey et al. 2000).

Table 4

5. Tissue classification

One of the most promising applications of gene expression analysis is the classification of tissue types according to their gene expression profiles (Golub et al. 1999). The power of gene expression analysis is directed at two important problems in this context: Cancer classification and drug discovery. Several recent studies (Alon et al.1999, Golub et al. 1999, Alizadeh et al. 2000) demonstrated that gene expression data can be used in distinguishing between similar cancer types, whose distinction is hard otherwise, thereby allowing more accurate diagnosis and treatment. Drug assessment is aided by expression profiling before, during and after treatment: The profiles pinpoint drug responsive genes, and indicate treatment outcome (Clarke et al. 1999).

Here we focus on application of gene expression analysis, and cluster analysis in particular, to cancer classification. In cancer related gene expression studies, the data consist of expression levels of thousands of genes in several tissues. The tissues originate from two or more known classes, e.g., normal and tumor. The analysis aims at studying the typical expression profile of each class and predicting the classification of new unlabeled tissues. Classification methods employ *supervised learning* techniques, i.e., the known classification of the tissues is used to guide the algorithm in building a classifier. These include support vector machines (Ben-Dor et al. 2000, Furey et al. 2000), boosting, clustering (Ben-Dor et al. 2000), discriminant analysis (Xiong et al. 2000) and weighted correlation (Golub et al. 1999). Classification can be aided by first filtering the dataset from genes which are irrelevant to the required distinction. Several methods have been suggested to choose subsets of *informative genes*, on which improved classification accuracy can be attained (Ben-Dor et al. 2000, Furey et al. 2000, Xiong et al. 2000, Dudoit et al. 2000).

Ben-Dor et al. (2000) have demonstrated the strength of clustering in classification problems. Key to their method is combining the *labeling* (known classification) information in the clustering process. Assume that we use a clustering algorithm with at least one free parameter. Given an unlabeled tissue, the clustering algorithm is applied repeatedly with different parameter values on the set of all tissues (known and unknown). Each solution is scored by its level of compatability with the labeling information, and the best solution is chosen. The classification of the unlabeled tissue is then determined according to the content of the cluster containing it, assigning it the most represented class in this cluster.

The *compatibility score* for a clustering solution used by Ben-Dor et al. (2001) is simply the number of tissue pairs that are mates or non-mates in both the true labeling and in the clustering solution. Singletons are considered as 1-member clusters for this computation. The clustering algorithm used in that work was CAST.

We have studied two classification datsets using CLICK. The first dataset of Alon et al. (1999) contains 62 samples of colon epithelial cells. These samples were collected from colon-cancer patients. They are divided into 40 'tumor' samples collected from tumors, and 22 'normal' samples collected from normal colon tissues of the same patients. Of the ~6000 genes represented in the experiment, 2000 genes were selected based on the confidence in the measured expression levels. This data is available at http://www.sph.uth.tmc.edu/hgc. The second dataset of Golub et al. (1999) contains 72 leukemia samples. These samples are divided into 25 samples of acute myeloid leukemia (AML) and 47 samples of acute lymphoblastic leukemia (ALL). Of the ~7000 genes represented in the experiment, 3549 were chosen based on their variability in the dataset. The complete dataset is available at http://www.genome.wi.mit.edu/MPR.

The application of CLICK to classify these datasets enumerates several combinations of parameters for CLICK, and chooses the one which is most compatible with the given labels. Compatibility was computed as in Ben-Dor et al. (2000). A sample is not classified if either it is a singleton in the clustering obtained, or no class has a majority in the cluster assigned to that sample.

In order to assess the performance of CLICK we employed the leave one out cross validation (LOOCV) technique, as done in (Ben-Dor et al. 2000). According to this technique, one trial is performed for each tissue in the dataset. In the *i*-th trial, the algorithm tries to classify the *i*-th sample based on the known classifications of the rest of the samples. The average classification accuracy is thus computed. Table 5 presents a comparison between the classification based on CLICK and that of CAST, as reported in Ben-Dor et al. (2000). The results are comparable, with CAST performing slightly better on the colon dataset, and CLICK performing slightly better on the leukemia dataset.

Table 5

Next, we tested CLICK's utility in differentiating between two very similar types of cancer. We concentrated on part of the leukemia dataset comprising of the 47 ALL samples only. For these samples an additional sub-classification into either T-cell or B-cell, is provided. An application of CLICK to this dataset resulted in an almost perfect classification (see Table 6).

Table 6

Finally we examined the influence of feature selection on the classification accuracy. To this end, we sorted the genes in each dataset according to the ratio of their between-sum-of-squares and within-sum-of-squares values, as suggested in (Dudoit et al. 2000). This ratio is computed by the following formula:

$$\frac{BSS(g)}{WSS(g)} = \frac{\sum_{i=1,2} n_i (x_{g,i} - x_g)^2}{\sum_{i=1,2} \sum_{k \in i} (x_g^k - x_{g,i})^2}$$

Here i denotes the class number, n_i its size, k denotes the sample number, $x_{g,i}$ is the average expression level of gene g at class i, x_g is the average expression level of gene g, and x_g^k is the expression level of gene g at sample k.

We chose the 50 genes with the highest value and performed the classification procedure on the reduced dataset which contained the expression levels of these 50 genes only. The results of this analysis are shown in Table 6. It is evident that in all cases performance was substantially improved on the reduced dataset. Moreover, there were no unclassified tissues in the reduced datasets.

Acknowledgments

We thank Amir Ben-Dor, Pablo Tamayo and Ralf Herwig for providing us with their clustering of the yeast cell-cycle dataset. The first author was supported by an Eshkol fellowship from the ministry of science, Israel. The second author was supported by a Joseph Sassoon fellowship. The third author was supported in part by the Israel Science Foundation (grant number 565/99).

References

Alizadeh A.A. , M.B. Eisen, R. Davis, C. Ma, I. Lossos, A. Rosenwald, J. Boldrick, R. Warnke, R. Levy, W. Wilson, M. Grever, J. Byrd, D. Botstein, P.O. Brown and L. M. Straudt (2000). Distinct types of diffuse large B-cell lymphomas identified by gene expression profiling. Nature 403:503-511.

Alon U., N. Barkai, D. A. Notterman, G. Gish, S. Ybarra, D. Mack, and A. J. Levine. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. PNAS, 96:6745-6750.

Ben-Dor A., L. Bruhn, N. Friedman, I. Nachman, M. Schummer, Z. Yakhini (2000). Tissue classification with gene expression profiles. Journal of Computational Biology 7(3/4):559-583.

Ben-Dor A., R. Shamir, and Z. Yakhini. (1999). Clustering gene expression patterns. Journal of Computational Biology, 6(3/4):281-297.

Brazma A. and J. Vilo (2000). Gene expression data analysis. FEBS Letters 480:17-24.

Cho R.J., M.J. Campbell, E.A. Winzeler, L. Steinmetz, A. Conway, L. Wodica, T.G.Wolfsberg et al (1998). A genome-wide transcriptional analysis of the mitotic cell cycle. Mol. Cell 2:65-73.

Clarke P.A., M. George, D. Cunningham, I. Swift, P. Workman (1999). Analysis of tumor gene expression following chemotherapeutic treatment of patients with bowel cancer. In proc. Nature Genetics Microarray Meeting 99, Scottsdale, Arizona, p. 39.

Coller H., C. Gradori, P. Tamayo, T. Colbert, E. Lander, R. Eisenman and T.R. Golub (2000). Expression analysis with oligonucleotide reveals that C-Myc regulates genes involved in growth, cell-cycle, signaling and adhesion. PNAS 97(7):3260-3265.

Cormack R.M.(1971). A review of classification (with discussion). J. Royal Statustical Society, Series A, 134:321-367.

Dudoit S., J. Fridlyand, T.P. Speed (2000). Comparison of discrimination methods for the classification of tumors using gene expression data. Technical report #576, Dept. of Statistics, university of California, Berkeley.

Eisen M.B. and P. O. Brown (1999). DNA arrays for analysis of gene expression. In Methods in Enzymology, Vol. 303, pages 179-205.

Eisen M.B., P. T. Spellman, P. O. Brown, and D. Botstein (1998). Cluster analysis and display of genome-wide expression patterns. PNAS, 95:14863-14868.

Even S. (1979). Graph Algorithms. Computer Science Press, Rockville, Maryland.

Everitt B. (1993). Cluster analysis. Edward Arnold, London, third edition.

Fodor S.P., R. P. Rava, X. C. Huang, A. C. Pease, C. P. Holmes, and C. L. Adams (1993). Multiplexed biochemical assays with biological chips. Nature 364:555-556.

Furey T.S., N. Cristianini, N. Duffy, D.W. Bendarski, M. Schummer, D. Haussler (2000). Support vector machine classification and validation of cancer tissue samples using microarray expression data. Bioinformatics 16:906-914.

Getz G., E. Levine, E. Domany, and M.Q. Zhang (2000). Super-paramagnetic clustering of yeast gene expression profiles. Physica, A279:457.

Golub T., D. Slonim, P. Tamayo, C.M. Huard, J.M. Caasenbeek, H. Coller, M. Loh, J. Downing, M. Caligiuri, C. Bloomfield, E. Lander (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286:531-537.

Golumbic M.C. (1980). Algorithmic Graph Theory and Perfect Graphs. Academic Press, New York.

Hansen P. and B. Jaumard (1997). Cluster analysis and mathematical programming. Mathematical Programming, 79:191-215.

Hao J. and J. Orlin (1994). A faster algorithm for finding the minimum cut in a directed graph. Journal of Algorithms 17(3):424-446.

Harkin D.P., J. Bean, D. Miklos, Y. Song, V. Maheswaram, J. Oliver, D. Haber (1999). Induction of GADD45 and JNK/SAPK-dependent apoptosis following inducible expression of BRCA1. Cell 97:575-586.

Harrington C.A., C. Rosenow, and J. Retief (2000). Monitoring gene expression using DNA microarrays. Curr. Opin. Microbiol., 3(3):285-291.

Hartigan J.A. (1975). Clustering Algorithms. John Wiley and Sons.

Hartuv E. and R. Shamir (2000). A clustering algorithm based on graph connectivity. Information Processing Letters, 76:175-181.

Herwig R., A.J. Poustka, C. Meuller, H. Lehrach, and J. O'Brien. Large-scale clustering of cDNA-fingerprinting data (1999). Genome Research 9(11):1093-1105.

Heyer L.J., S. Kruglyak, and S. Yooseph (1999). Exploring expression data: identification and analysis of coexpressed genes. Genome Research, 9(11):1106-1115.

Hughes J.D., P.E. Estep, S. Tavazoie and G.M. Church (2000). Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae. J. Mol. Biol. 296:1205-1214.

Iyer V.R., M.B. Eisen, D.T. Ross, G. Schuler, T. Moore, J.C.F. Lee, M. Trent, L.M. Staudt, J. Hudson, M.S. Boguski, D. Lashkari, D. Shalon, D. Botstein, and P.O. Brown (1999). The transcriptional program in the response of human fibroblast to serum. Science 283:83-87.

Jelinsky S.A., P. Estep, Q.M.Church and L.D.Samson (2000). Regulatory networks revealed by transcriptional profiling of damaged Saccharomyces cerevisiae cells: Rpn4 links base excision repair with proteasomes. MCB 20(21):8157-8167.

Kerr M.K., M. Martin, and G.A. Churchill (2000). Analysis of variance for gene expression microarray data. Technical report, The Jackson Laboratory.

Kohonen T.(1997). Self-Organizing Maps. Springer, Berlin.

Lance G.N. and W.T. Williams (1967). A general theory of classification sorting strategies. 1. hierarchical systems. The Computer Journal, 9:373-380.

Lipshutz R.J., S. P. A. Fodor, T. R. Gingeras, and D. J. Lockhart (2000). High density synthetic oligonucleotide arrays. Nature Genetics Supplement, 21:20-24.

Livesey F.J., T. Furukawa, M.A. Steffen, G.M. Church and C.L. Cepko (2000). Microarray analysis of the transcriptional network controlled by the photoreceptor homeobox gene Crx. Curr. Biol. 10:301-310.

Maleck K., A. Levine, T. Eulgem, A. Morgan, J. Schmid, K. A. Lawton, J.L. Dangl and R.A. Dietrich (2000). The transcriptome of Arabidopsis thaliana during systematic acquired resistance. Nature Genetics 26:403-410.

Marshall A. and J. Hodgson (1998). DNA chips: an array of possibilities. Nat Biotechnol, 16:27-31.

Milosavljevic A., Z. Strezoska, M. Zeremski, D. Grujic, T. Paunesku, and R. Crkvenjakov (1995). Clone clustering by hybridization. Genomics, 27:83-89.

Mirkin B. (1996). Mathematical Classification and Clustering. Kluwer.

Poustka A.J., R. Herwig, A. Krause, S. Hennig, S. Meier-Ewert, and H. Lehrach (1999). Toward the gene catalogue of sea urchin development: the construction and analysis of an unfertilized egg cDNA library highly normalized by oligonucleotide fingerprinting. Genomics, 59:122-133.

Ramsay G.(1998). DNA chips: State-of-the art. Nat Biotechnol, 16:40-44.

Roth F.P., J.D. Hughes, P.W. Estep, and G.M. Church (1998). Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. Nature Biotech. 16:939-908.

Schena M.(1996). Genome analysis with gene expression microarrays. Bioessays, 18:427-431.

Schena M., D. Shalon, R. Heller, A. Chai, P. O. Brown, and R. W. Davis (1996). Parallel human genome analysis: microarray-based expression monitoring of 1000 genes. PNAS 93:10614-9.

Sharan R. and R. Shamir (2000). CLICK: A clustering algorithm with applications to gene expression analysis. In Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology (ISMB), pages 307-316.

Shamir R. and R. Sharan (2001). Algorithmic approaches to clustering gene expression data. In Current topics in computational biology, T. Jiang, T. Smith, Y. Xu and M.Q. Zhang eds., MIT Press.

Spellman P.T., G. Sherlock, M. Zhang, V.R. Iyer, K. Anders, M. Eisen, P.O. Brown, D. Botstein and B. Futcher (1998). Comprehensive identification of cell cycle regulated gene of the yeast Saccharomyces Cerevisia by microarray hybridization. Mol. Biol. Cell 9:3273-3297.

Tamayo P., D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E. S. Lander, and T.R. Golub (1999). Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. PNAS, 96:2907-2912.

Tavazoie S., J. Hughes, M. Campbell, R. Cho and G. M. Church (1999). Systematic determination of genetic network architecture. Nature Genetics 22:281-285.

Toronen P., M. Kolehmainen, G. Wong, and E. Castren (1999). Analysis of gene expression data using self-organizing maps. FEBS Letters, 451:142-146.

Werner T. (2001). Target gene identification from expression array data by promoter analysis. Biomolecular Engineering 17:87-94.

Xiong M., L. Jin, W. Li, E. Boerwinkle (2000). Computational methods for gene expression based tumor classification. Biotechniques 29:1264-1270.

Yeung K.Y., D.R. Haynor, and W.L. Ruzzo (2001). Validating clustering for gene expression data. Bioinformatics 17:309-318.

Zhang M.Q (1999). Large scale gene expression data analysis: a new challenge to computational biologists. Genome Research 9:681-688.

Zhao R., K. Gish, Y. Yin, D. Notterman, W. Hoffman, E. Tom, D. Mak and A.J. Levine (2000). Analysis of p53 regulated gene expression patterns using oligonucleotide arrays. Genes and Dev. 14:981-993.

Zhu J. and M.Q. Zhang (1999). SCPD: a promoter database of the yeast Saccharomyces cerevisiae. Bioinformatics 15:607-611.

Table Legends

Table 1: A comparison between CLICK and GeneCluster on the yeast cell-cycle dataset of (Cho et al. 1999).

Table 2: A Comparison between CLICK and the hierarchical clustering of Eisen et al. (1998) on the dataset of response of human fibroblasts to serum of (Iyer et al. 1999).

Table 3: A summary of the clustering solutions and their figures of merit on the yeast cell-cycle dataset of (Spellman et al. 1998).

Table 4: (A) Statistics on motifs identified in CLICK's clusters (first column), GeneCluster's clusters (second column) and in both clusterings (third column). First row: total number. Second row: number of motifs with a hit in the SCPD DB. (B) Motifs identified in both CLICK's and GeneCluster's clusters. Left column: CLICK's motif name. Each motif is denoted by two numbers: the first is the cluster number and the second is the motif serial number in the cluster (AlignACE is capable of finding multiple motifs in a target set of sequences by an iterative masking procedure). The second row contains statistics on the prevalence of the motif in the cluster. Second column: the motif's consensus sequence. Third column: an experimentally verified TF which matches the consensus sequence. (For TFs denoted by an '*', there is one mismatch between the TF binding site consensus and the identified motif consensus, otherwise, there is a perfect match.) Forth column: The corresponding GeneCluster's motifs (motifs with similarity coefficient above 0.7 to CLICK's consensus). Motifs found more than once are grouped together in column 1.

Table 5: A comparison of classification quality between CLICK and CAST on the colon data of (Alon et al. 1999) and the leukemia data of (Golub et al. 1999). For each dataset and clustering algorithm the percent of correct classifications (in the LOOCV iterations), incorrect classifications and unclassified elements are specified.

Table 6: A summary of the classifications obtained by CLICK on the colon data of (Alon et al. 1999), the whole leukemia dataset of (Golub et al. 1999), and part of the leukemia dataset which contains ALL samples only. For each dataset classifications were performed with respect to the total number of genes, and with respect to the 50 most informative genes. The percent of correct classifications (in the LOOCV iterations), incorrect classifications and unclassified elements are specified.

Figure Legends

Figure 1:CLICK's clustering of the fibroblasts serum response dataset of (Iyer et al., 1999). x axis: 1-12: synchronized time points. 13: unsynchronized point. y axis: normalized expression levels. The solid line in each sub-figure plots the average pattern for that cluster. Error bars display the measured standard deviation. The cluster size is printed above each plot.

Tables

Table 1:

Program	#Clusters	Homogeneity	Separation
CLICK	30	0.8	-0.07
GeneCluster	30	0.74	-0.02

Table 2:

Program	#Clusters	Homogeneity	Separation
CLICK	10	0.88	-0.34
Hierarchical	10	0.87	-0.13

Table 3:

Program	#Clusters	Homogeneity	Separation
CLICK	6	0.66	-0.1
K-Means	49	0.63	0.09
GeneCluster	6	0.62	-0.07
CAST	5	0.6	-0.15
‘True’	5	0.57	-0.13

Table 4:

4A.

	CLICK	***GeneCluster***	***Common***
Found	26	30	17
Verified	17	19	13

4B.

CLICK	**Consensus**	**Putative TF**	**GeneCluster**
Clk 1_2 16% (37/237)	GGTGGCAAAW	UASPHR	Som 14_2 30% (61/205)
Clk 2_1 63% (119/189)	RAAAAAAAAA	PHO2;SWI5	Som 4_4 38% (52/136)
Clk 2_2 44% (83/189)	ATGTAYGGRTK	RAP1	Som 1_2 52% (85/165)
Clk 2_3 30% (56/189) Clk 2_11 23% (44/189) Clk 9_3 87% (33/38)	RAAAAATTT AAAAAWTTT TGAAAAWTTTT	DAL82	Som 2_2 65% (48/74) Som 4_2 61% (83/136)
Clk 2_7 15% (29/189)	NSYAGGCNGNR	RAP1*, BUF*	Som 1_4 17% (28/165) Som 1_8 11% (18/165) Som 1_9 15% (25/165)
Clk 2_9 12% (23/189)	CYCNSCNRGNNGGA	MCM1*	Som 1_5 15% (25/165)
Clk 3_3 21% (31/149)	YNCGGNSNNNSGGS	RAP1*	Som 3_8 13% (23/175)
Clk 3_4 19% (28/149)	RRCCAATCAN	ABF1,BAF1*	Som 3_2 21% (36/175)
Clk 3_6 12% (18/149) Clk 3_7 8% (12/149)	GGCNGGGCRKC TSGGCGGCNNTT	URS1H	Som 3_4 8% (14/175)
Clk 3_9 19% (28/149)	SGGNNNNNNNGGNN NGG	BUF *	Som 3_6 16% (28/175)
Clk 2_8 15% (28/189)	SCNGCNNSCNGNNGS G	-----	Som 1_6 17% (28/165)
Clk 3_11 13% (19/149)	MNNNGGGNNNRNNN RNGGGR	-----	Som 6_7 25% (28/114)
Clk 3_21 9% (13/149)	NCCGNYGGNCCGR	-----	Som 3_8 13% (23/175)
Clk 26_2 90% (9/10)	AGGGGCGGNG	-----	Som 9_3 15% (23/150)

Table 5:

Dataset	Method	Correct	Incorrect	Unclassified
Colon	CLICK	85.5	9.7	4.8
	CAST	88.7	11.3	0.0
Leukemia	CLICK	90.3	4.2	5.5
	CAST	87.5	12.5	0.0

Table 6:

Dataset	Size	Correct	Incorrect	Unclassified
Colon	2000	85.5	9.7	4.8
	50	90.3	9.7	0.0
Leukemia	3549	90.3	4.2	5.5
	50	98.6	1.4	0.0
ALL	3549	95.8	2.1	2.1
	50	100.0	0.0	0.0