Proceedings of the 2001 IEEE Congress on Evolutionary Computation Seoul, Korea • May 27-30, 2001

Evolution Strategy Applied to Global Optimization of Clusters in Gene Expression Data of DNA Microarrays

Kwonmoo Lee[1], Ju Han Kim[2], Tae Su Chung[1], Byoung-Sun Moon[1], Hoseung Lee[1], Isaac S. Kohane[2]
[1]Bioinformatics Laboratory, Information Technology R&D Center, Samsung SDS, Seoul, Korea, 135-918
[2]Children's Hospital Informatics Program, Children's Hospital, Harvard Medical School, Boston, MA, 02115

Abstract- Cluster analysis is the most important method for analyzing large-scale gene expression patterns. The matrix representation of microarray data and its successive 'optimal' incisional hyperplanes that create top-down hierarchical tree are a useful platform for developing optimization algorithms to determine the 'optimal' clusters from a pairwise proximity matrix which represents completely connected and weighted graph [1]. Evolution strategy is applied to determine the 'globally optimal' incisional hyperplanes to construct hierarchical tree structure and tested with Fisher's iris and Golub's leukemia data sets. The results were compared with those of bottom-up hierarchical clustering, K-means and SOMs(Self-Organizing Maps) algorithms with promising results.

1 Introduction

The unprecedented high-throughput of DNA sequence information by various sequencing projects such as Human Genome Project requires new paradigm for biomedical research. Even though deciphering human genome has been finished, we still do not know what the human genome tells us. Functional genomics aims to reveal cooperative roles of genes at a genome-wide scale as well as the biological functions of each gene. This will lead to understanding the mechanism of cell development, causes of diseases, and effects of drug target, etc.

The DNA microarray is a powerful experimental tool for extracting functional information from genome [2, 3]. On glass surface, cDNA (complementary DNA) or oligonuleotide fragments for hybridization are regularly arrayed in high density. The genes, the protein-encoding DNA regions of genome, play biological roles when they are translated into protein via mRNA transcription, which is called gene expression. The DNA microarrays enable large-scale measurement of expressed mRNA from living cells, thereby we can investigate gene expression patterns which are very important clues for understanding biological functions of the corresponding genes. Especially, the oligonucleotide microarrays can be used to detect sequence polymorphisms or mutations in genomic DNA. They are very promising to be applied for diagnosis of disease in a molecular level.

The schematic procedures for monitoring gene expression using DNA microarrays are illustrated in Figure 1. After mRNA is extracted from tissues of two different conditions (e.g. normal and cancer cells), it is reversely transcribed to DNA, called cDNA. The cDNA's from two different tissues are labeled with fluorescent dyes of different colors (red and green) and bound by the base paring (A-T, G-C) to the spot of the microarray, made of the complementary sequence cDNA, that is called hybridization. After unbounded cDNA is washed off, the microarray is scanned by green and red lasers. We can compare the quantity of the each gene expression in the two conditions after some image processing. We can mine functional information of each gene by analyzing the large-scale gene expression profiles obtained by the microarray experiments in various kinds of situations such as drug treatments, cancer types, stages of cell development, etc.

Cluster analysis is the most important method for analyzing DNA microarray data. It extracts internal structure from observed data by dividing them into meaningful groups. It is the starting point to explore the internal structure of gene expression data in DNA microarrays. Because the genes of similar expression profiles may share similar functions, clustering gene expression data in *Saccharomyces cerevisiae* grouped the genes of known similar functions and have also shown to be used for tentative assignment of functional annotation of unknown genes based on known genes [4]. Cluster analysis is also useful in classifying tissues on the basis of gene expression. The cancerous and normal tissues are distinguished even with genes of subtle differences in gene expression [5]. Golub *et al.* proposed a class discovery procedure using cluster analysis [6]. The procedure categorizes leukemia into acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) without prior knowledge. They also developed leukemia class predictor by supervised learning method based on gene expression data. In addition, co-regulated genes can be identified by cluster analysis, since there is the correlation between sequence motifs in promoter region and gene expression profile [7, 8]. Holmes *et al.* considered gene expression data and promoter sequences simultaneously to find co-regulated genes by cluster analysis [9].

In Sec. 2, we will briefly review the widely used clustering algorithms in analysis of DNA microarray data, such as bottom-up hierarchical clustering, K-means, and self-organizing maps (SOMs). In Sec. 3, global optimization approach for cluster analysis using evolution strategy will be described. In Sec. 4, we will demonstrate the performance of the algorithm with Fisher's iris and Golub's leukemia data sets by comparing with the results of other clustering algorithms. This paper will be concluded in Sec. 5.

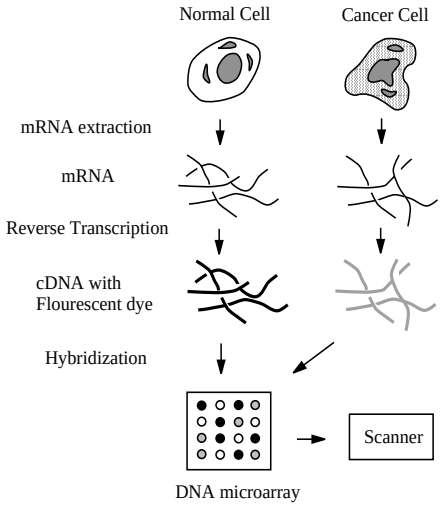

Figure 1: Schematic procedure of DNA microarray experiments

2 A Brief Review of Cluster Algorithms

In this section, we will give a brief review of widely-used cluster algorithms: Bottom-up hierarchical clustering, K-means, and SOMs. The bottom-up hierarchical clustering algorithm is the first algorithm applied to analyze the gene expression data in DNA microarrays [4]. The goal of the algorithm is to build phylogenetic tree(dendrogram) which graphically represents hierarchy of nested clusters. The algorithmic procedure is as follows:

(1) Compute the dissimilarity(distance) matrix, if necessary,

(2) Find the closest pair of elements or clusters and merge them,

(3) Update the dissimilarities between combined cluster and the others,

(4) Repeat steps (2) and (3) until only one cluster remains.

Input parameter of this algorithm is the scheme to modify the dissimilarity matrix in step (3). Single-linkage, complete-linkage and average-linkage methods are common variants of this scheme. In a single/complete/average link version, the dissimilarity of two clusters is defined by the minimum/maximum/average of dissimilarities between any pairs of elements in different clusters. Even though it is quite efficient, this algorithm is not robust, and the global feature of cluster structure is not considered but clusters agglomerate with local relation between clusters

The K-means cluster algorithm is very simple. The goal of the algorithm is to divided data set into k clusters to minimize the error function defined to be sum of distances between centroids and their associated elements [10]. The number of cluster, k, is given or known. The algorithmic procedure is as follows:

(1) Choose k elements as initial centroids,

(2) Assign all elements to the nearest centroid,

(3) Update the centroid of each cluster,

(4) Repeat steps (2) and (3) until the centroids are fixed or pre-assigned number of iteration is reached.

The key issues of this algorithm are how to select initial centroids in step (1) and how to update the centroid in step (3). Random selection is simple but it often produces poor results. So, other techniques are required to improve this algorithm. Usually, the new centroid is taken as a mean vector of elements in the cluster. But, if we take another reasonable error function, then we should take the median of the elements. It has difficulties in analyzing the data whose clusters are overlapped and outliers exist. In addition, it is restricted to the data in Euclidean space.

The SOMs uses a type of unsupervised learning based on neural network [11]. The goal of the algorithm is to give topology-preserving map from high-dimensional input data into a feature map of a low-dimensional (usually 1 or 2 dimensional) output. The algorithmic procedure is as follows:

(1) Initialize the weight matrix and parameters,

(2) For each input element, determine the "winning" output node with minimum distance and update weights to winning node and its neighbors,

(3) Repeat step 2 until the weight matrix converges or pre-assigned number of iteration is reached.

The input parameters of this algorithm are the size of output nodes, the initialization algorithm of weight matrix and the strategy of updating the matrix. It is known to be very fast and effective for visualizing the results, while the disadvantages are that the output topology is predefined and results depend on initialization and learning rate.

3 Evolution Strategy Applied to Cluster Analysis

3.1 Data Representation

The DNA microarray data for clustering can be represented by the completely connected weighted-graph with similarity measure. The vertex of the graph corresponds to each object to be clustered. The edge represents the similarity between objects. The similarity measures can be stored in pairwise proximity matrix. This representation makes the algorithm independent of similarity measures, differing from K-means and SOMs.

3.2 Clustering as an Optimization Problem

Our clustering scheme is successive binary partitioning, which produces top-down hierarchical tree. Binary partitioning can be done by the incisional hyperplane which de-

compose the graph into two parts with optimized figure of merit (clustering index) as illustrated in Figure 2. Kim *et al.* proposed some figure of merit called MII (Matrix Incision Index), which includes homogeneity and separation of binary partition [1]. Homogeneity indicates that each object within the same cluster should have high similarity. Separation means that the objects between clusters should have low similarity. These requirements are incorporated into MII as follows:

$$\text{MII} = \frac{(m/(n+m)) * b + (n/(n+m)) * c}{a} \qquad (1)$$

where m and n are sizes of groups 1 and 2, respectively, a is the average link strength between groups 1 and 2, b and c are within-group average link strength of group 1 and 2, respectively. The numerator corresponds to homogeneity, the denominator to separation. Since homogeneity/separtion should be as high/low as possible in binary partitioning, we should maximize the MII. The index is defined directly from the similarity matrix without prior information regarding the structure of data set.

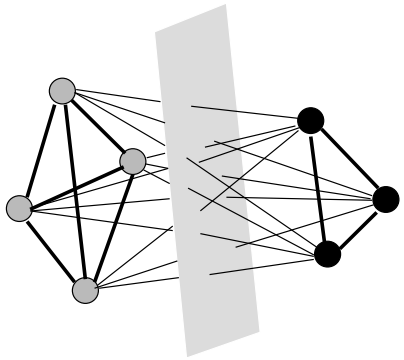

Figure 2: The incisional hyperplane decomposes completely connected graph into two sub graphs

After we define the MII as a good clustering index, we should find how to get to global maximum of the MII. Because there is no general and rigorous mathematical way for the global optimization problem, a feasible way is to use heuristic methods to adopt randomness to escape local optima, one of which is evolution strategy used in this paper.

The advantages of global optimization of clustering is to increase clustering quality because the clustering index is directly optimized differing from other algorithms. In addition, we do not have any prior assumption about data structure. The possible disadvantage is that this algorithm might be relatively slower than other greedy algorithms such as K-means and SOMs.

3.3 Evolution Strategy to Find Globally Optimized Clusters

Although global maximization of MII gives the best binary partition in some sense, the order of computation for finding global maximum grows exponentially to the size of data set. So, we need a heuristic algorithm to find the global maximum. Evolution strategy (ES) is an effective search algorithm in optimization problems which simulates the biological evolution. We applied ES to optimize the MII as follows:

First, consider the sequence $\{x_i\}$ with $0 \le x_i \le 1$ which represents the membership rate of data set. If the membership rate x_i is close to zero/one, the ith object has great possibility to belong to the first/second cluster. By the membership sequence $\{x_i\}$, the MII (Eq. 1) can be written down with the following equations.

$$m = \sum_i (1 - x_i) \quad \text{and} \quad n = \sum_i x_i \qquad (2)$$

$$a = \sum_{i \neq j} L_{ij}(1 - x_i)x_j \qquad (3)$$

$$b = \sum_{i<j} L_{ij}(1 - x_i)(1 - x_j) \qquad (4)$$

$$c = \sum_{i<j} L_{ij}x_i x_j \qquad (5)$$

where L_{ij} is the similarity of objects indexed by i and j.

Once we define MII as the function of the membership sequence, we applied standard evolution strategy to find globally optimized binary partitioning represented by membership sequence $\{x_i\}$. We used (15,100)-ES, which means that the numbers of parents and offsprings are 15 and 100, and the parents in the next generation are selected out of the offsprings in the current generation. The types of recombination and mutation in this optimization are local intermediary recombination and Cauchy mutation. The local intermediary recombination works by selecting two parents and calculating a weighted sum of the components of the two parents. The Cauchy mutation has been shown to performs better than Gaussian mutation because of higher probability of making longer jumps [12]. The individual mutation rates with local intermediary recombination are used in this computation.

4 Results

In this section, we will show the clustering results of two public data sets, Fisher's iris [13] and Golub's leukemia gene expression data sets [6] by analyzing the algorithm presented in this paper. In order to assess the performance of the clustering, the results will also be compared with those of publicly available softwares. We have used CLUSTER written by Stanford University, in which bottom-up hierarchical clustering, K-means algorithm, and SOMs have been implemented (http://rana.stanford.edu/software/). GENECLUSTER written by Whitehead Institute of Biomedical Research [14] was also used for SOMs (http://www-genome.wi.mit.edu/MPR/software.html). In order to compare the reproducibility of each algorithm, we carried out each algorithm ten times and evaluate the standard deviation of clustering errors.

4.1 Fisher's Iris Data

The Fisher's iris flower data set has been widely used for evaluation of cluster algorithms. The iris data set consists of four measurements (petal and sepal length and petal and sepal width) of 50 *Iris Setosa*, 50 *Iris Versicolor*, and 50 *Iris Virginica*. In the test, the similarity measures between the iris flowers are the absolute values of the Pearson's correlation coefficient which will be used for all tests of our ES and bottom-up hierarchical clustering. As illustrated in Figure 3, the first binary partitioning separated all *Setosa* from the entire data set without any error. The next partitioning of 50 *Versicolor*'s and 50 *Virginica*'s generated six errors. Three *Versicolor*'s (objects 9, 12, 40) were clustered to *Virginica* and three *Virginica*'s (objects 66, 77, 81) to *Versicolor*. The overall accuracy of clustering was 96% (144/150). The repeated results of our algorithm were the same.

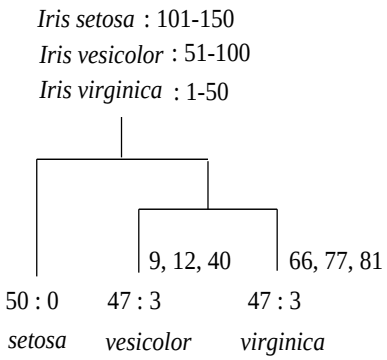

Figure 3: Clustering result of Fisher's iris flowers by ES approach. The lists of numbers in the terminal leaves are those of misclassified objects

The bottom-up hierarchical clustering in CLUSTER with the same iris data have generated wide range of errors, depending upon the type of clustering, complete, average, and single linkage clustering. In the three types of clustering, all *Setosa*'s were separated without errors. However, in clustering *Versicolor* and *Virginica*, 22 errors (92% accuracy) were generated in complete linkage clustering, 8 errors (94.7% accuracy) in average linkage clustering. The single linkage clustering did not separate the two iris species in the second highest branch. The repeated results of each type were the same, respectively.

The K-means algorithm implemented in CLUSTER has shown clustering performance with iris data as good as our clustering algorithm. We used two strategies of clustering, one of which is to make three partition of the data set. The other is successive binary partitioning, the top-down hierarchical method as used in our algorithm. In tri-partitioning, the mean number of errors was 5.8 (96.1% accuracy), and the standard deviation was 0.9. Most of errors existed in group of *Versicolor* and *Virginica* while *Setosa* was separated with 0 or 1 error. In successive binary partitioning, the mean number of errors was 5.4 (96.4% accuracy), and the standard deviation was 2.2. We had no errors in separating *Setosa*.

We tested two SOMs implemented in CLUSTER and GENECLUSTER with two strategies as used in testing K-means. By SOMs in CLUSTER, tri-partitioning method gaves rise to 5 errors (95.3% accuracy) in all repeated clustering. Even though successive binary partitioning sometimes led to only 2 errors (98.7% accuracy) at best, in most of the repeated results there was no partitioning between *Versicolor* and *Virginica* by 1×2 SOM. The SOMs in GENCLUSTER produced lower performance with Iris data. In tri-partitioning, the mean number of errors was 16.3 (89.1% accuracy), and the standard deviation was 0.5. In binary successive partitioning, we had 19 errors (87.3% accuracy) with the same repeated results.

4.2 Golub's Leukemia Data

We analyzed Golub's leukemia data set from 38 human acute leukemia cells, which was used for training class predictor of acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) [6]. The ALL group consists of T-cell ALL's (T-ALL) and B-cell ALL's (B-ALL). In the experiment, RNA extracted from each leukemia sample was hybridized to high density microarrays containing 6,817 human genes. After scanning the microarrays and image processing, the expression levels of each gene for each leukemia cell were measured. It has been shown that leukemia class prediciton was feasible by this gene expression monitoring without additional biological knowledge. Golub *et al.* selected 50 genes highly correlated with ALL-AML class distinction. The Figure 4 demonstrates the results of our cluster algorithm applied to leukemia data using the 50 genes. The first binary partitioning distinguished ALL from AML with 1 error, and the second distinction of ALL between T-ALL and B-ALL gave rise to 1 error. Two B-ALL samples were misclassified into AML and T-ALL, respectively. The overall accuracy of the result was 94.7% (36/38). The repeated results of our algorithm were the same. Even if the 50 genes had been selected for only AML-ALL classification, the B-ALL and T-ALL samples were successfully separated because even minute differences in gene expression levels can be used in cluster analysis as indicated by Alon *et al.* [5].

The bottom-up hierarchical clustering with complete linkage produced 3 errors (92.1% accuracy) where 2 errors were in AML-ALL distinction and 1 error in partitioning T-ALL and B-ALL. On the other hand, average and single linkage clustering did not distinguish T-ALL from B-ALL at the second highest tree branch, even if they succeeded in ALL-AML distinction with 1 or 2 errors. The repeated results of each type were the same, respectively.

In tri-partitioning by K-means in CLUSTER, the mean number of errors was 7.7 (79.7% accuracy), and the standard deviation was 4.9. But, the successive binary partitioning scheme increased the clustering performance with 3.6 average errors (90.5% accuracy) and 0.5 standard deviation.

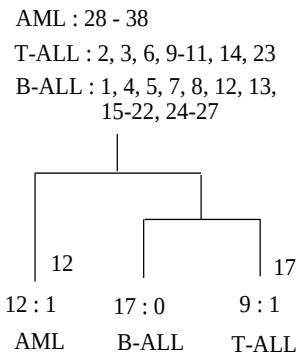

Figure 4: Clustering result of Golub's leukemia data by ES approach. The lists of numbers in the terminal leaves are those of misclassified objects

The SOMs of CLUSTER and GENECLUSTER gave different results as in iris data. By SOMs in CLUSTER, we had 10 errors (73.7% accuracy) with the same repeated results in tri-partitioning, and 2 errors (94.7% accuracy) at best in successive binary partitioning while the distinction between T-ALL and B-ALL had not been achieved three times in binary partitioning. The SOMs of GENECLUSTER in both clustering schemes did not distinguish T-ALL from B-ALL quite well while AML-ALL distinction was obtained with 1 or 2 errors. The mean number of overall errors was 11.8 (68.9% accuracy) and the standard deviation 1.6 in tri-partitioning. In successive binary partitioning, we had 11.7 average errors (69.2% accuracy) and 3.0 standard deviation .

As summarized in Table 1, clustering accuracy of our ES clustering with iris and leukemia data sets is acceptable in comparison with other algorithms. In addition, the repeated results of our algorithm were persistent while K-means and SOMs produce variable results, and SOMs sometimes did not partition iris and leukemia data. However, the weakness of this approach is that it requires much more computational cost than other algorithms. Especially, the objects which do not inherently belong to evident groups make convergence much slower. The sophisticated handling of these objects may speed up the algorithm.

5 Conclusion

Due to the recent development of high-throughput experimental techniques in functional genomics, we are facing the flood of large-scale gene expression data where various functional information of biological entities is inherent. The needs in analysis of such biological data sheds light on the importance of cluster analysis that is basic methodology to reveal internal structure of complex data set. Because cluster analysis is the first phase of mining useful information, the reliability of clustering results is responsible for the quality of information extracted in the following stages.

The direct maximization of the figure of merit (clustering index) which is main difference with conventional cluster algorithms can be good strategy to increase clustering quality. We have shown that evolution strategy was effectively applied to find globally optimal clusters. In addition, our approach does not needs any prior assumption about data structure and considers global perspective of data in clustering. We can also apply this algorithm with different clustering indices and similarity measures, which is not possible in algorithm-dependent methods such as K-means and SOMs.

Even though the computational cost is quite high in this preliminary study with iris and leukemia data sets, we have shown that our approach is promising for the further investigation with a variety of data sets, similarity measures and optimization algorithms.

Clustering Algorithms			Errors (μ+σ)	
			Iris	Leukemia
Evolution Strategy Clustering			6+0	2+0
Bottom-up Hierarachical Clustering	complete linkage		22+0	3+0
	average linkage		8+0	N.P.
	single linkage		N.P.	N.P.
K-means	tri-partitioning		5.8+0.9	7.7+4.9
	succesive binary partitioning		5.4+2.2	3.6+0.5
SOMs	CLUSTER	tri-partitioning	5+0	10+0
		succesive binary partitioning	2 or N.P.	2 or N.P.
	GENECLUSTER	tri-partitioning	16.3+0.5	11.8+1.6
		succesive binary partitioning	19+0	11.7+3.0

N.P. : No Partition μ : mean σ : standard deviation

Table 1: Summary of clustering results

Bibliography

[1] J. H. Kim, L. Ohno-Machado, and I. S. Kohane, "Unsupervised learning from complex data: The Matrix incision tree algorithm", Pacific Symposium on Biocomputing, 2001; 30-41.

[2] P. O. Brown and D. Botstein, "Exploring the new world of the genome with DNA microarrays", Nature, 1999; 21: 33-7.

[3] D. J. Lockhart and E. A. Winzeler, "Genomics, gene expression and DNA arrays", Nature, 2000; 405: 827-36.

[4] M. B. Eisen, P. T. Spellman, P. O. Brown, D. Botstein, "Cluster analysis and display of genome-wide expression patterns", Proc. Natl. Acad. Sci. USA, 1998; 95: 14863-8.

[5] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine, "Broad patterns of gene expression revealed by clustering analysis of tumor

and normal colon tissues probed by oligonucleotide arrays", Proc. Natl. Acad. Sci. USA, 1999; 96: 6745-50.

[6] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Caasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, E.S. Lander, "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring", Science, 1999; 286: 531-7.

[7] S. Tavazoie and G. M. Church, "Quantitative whole-genome analysis of DNA-protein interactions by in vivo methylase protection in *E. coli* ", Nature Biotechnol., 1998; 16:566-71.

[8] S. Tavazoie, J. D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church, "Systematic determination of genetic network architecture", Nature Genetics, 1999; 22: 281-5.

[9] I. Holmes and W. J. Bruno, "Finding regulatory elements using joint likelihoods for sequence and expression profile data", Intelligent System for Molecular Biology, 2000; 202-10.

[10] J. A. Hartigan, "Clustering Algorithms', 1975, Wiley, New York.

[11] T. Kohonen, "Self-Organizing Maps", 1997, Springer, New York.

[12] X. Yao, Y. Liu, and G. Lin, "Evolutionary Programming Made Faster", IEEE Trans. on Evol. Comp., 1999;2:82-102.

[13] R. A. Fisher, "The use of multiple measurements in taxonomic problems", Annals of Eugenics, 1936, 7, Part II: 179-88.

[14] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E. S. Lander, T. R. Golub, "Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation", Proc. Natl. Acad. Sci. USA, 1999;96:2907-12.