A Survey on Genetic Algorithm based Clustering Techniques for Micro array gene data

K.Vivekanandan [1] P.Krishnakumari [2]

Abstract

Clustering is a key step in the analysis of gene expression data, and in fact, many classical clustering algorithms are used, or more innovative ones have been designed and validated for the task. Despite the widespread use of artificial intelligence techniques in bioinformatics and, more generally, data analysis, there are very few clustering algorithms based on the genetic paradigm, yet that paradigm has great potential in finding good heuristic solutions to a difficult optimization problem such as clustering. In this paper the nature of microarray data is discussed briefly and a survey on genetic algorithm based clustering techniques for micro array gene data is presented. Some preliminary concepts that form the basis for the development of clustering algorithms are introduced. Finally, some of the most popular clustering techniques like GenClust, HGACLUS, hybrid method using EM algorithm, multiobjective genetic clustering algorithm are discussed. As such, the study provides a framework for the evaluation of clustering in gene expression analysis.

Keywords: clustering, gene data, genetic algorithm, microarray data analysis

[1] Reader , Bharathiar University, Coimbatore, Tamilnadu, India.

[2] Senior Lecturer, Department of Computer Science, Sri Ramakrishna College of Arts and Science for Women , Coimbatore -641044 , Tamilnadu, India.

E-mail:kkjagadeesh@yahoo.com

1. Micro Arraydata and its Complexity

The recently developed gene expression micro array technique measures the expression levels of thousands of genes in a single experiment. This large amount of data is something of a gold mine, from which a number of things can be found. Gene-expression micro array data have been explored in a variety of ways including gene clustering, gene selection and many others. Gene expression micro arrays are a prominent experimental tool in functional genomics which has opened the opportunity for gaining global, systems-level understanding of transcriptional networks. Micro array platforms for measuring the expression levels of most or all genes of an organism are available for a variety of organisms ranging from yeast to human. Experiments that use this technology typically generate overwhelming volumes of data, unprecedented in biological research, which makes the task of mining meaningful biological knowledge out of the raw data a major challenge. Hence, exploitation of gene expression data is fully dependent on the availability of advanced data analysis and statistical tools. Many clustering [9,25] algorithms and software tools for analysis of microarray data were developed in recent years. Clustering algorithms applied to gene expression data partition the genes into distinct groups according to their expression patterns over the probed biological conditions. Such partition should assign genes with similar expression patterns to the same cluster (keeping the *homogeneity* merit of the clustering solution) while retaining the distinct expression pattern of each cluster (ensuring the *separation* merit of the solution). Cluster analysis eases

the interpretation of the data by reducing its complexity and revealing the major patterns that underlie it. Clustering is the task of organizing a set of objects into meaningful groups. These groups can be disjoint, overlapping, or organized in some hierarchical fashion. The key element of clustering is the notion that the discovered groups are meaningful. Clustering is an exploratory tool for analyzing large datasets, and has been extensively used in numerous application areas. Clustering has a wide range of applications in life sciences and over the years it has been used in many areas ranging from the analysis of clinical information, phylogeny, genomics, and proteomics. For example, clustering algorithms applied to gene expression data can be used to identify co-regulated genes and provide a genetic fingerprint for various diseases. The primary goal of this article is to provide an overview of the various issues involved in clustering large datasets, describe the merits and underlying assumptions of some of the commonly used clustering approaches, and provide insights on how to cluster datasets based on genetic algorithm paradigm. The article is organized as follows. The sections 2 to 5 describe the various types of clustering algorithms developed over the years, similarity measures, limitations of the conventional clustering algorithms and dimensionality reduction. The sections 6 and 7 focus on genetic algorithm, criteria for evaluating clustering algorithms. The section 8 describes microarray technology and focuses on the problem of clustering data arising from microarray experiments. Finally, section 9 provides a brief introduction to the GA based clustering techniques like GenClust, HGACLUS, hybrid method using EM algorithm, multiobjective genetic clustering algorithm.

2. Types Of Clustering Algorithms

The topic of clustering has been extensively studied in many scientific disciplines and a variety of different algorithms have been developed [18, 30, 31]. Two recent surveys on the topics [15, 17] offer a comprehensive summary of the different applications and algorithms. These algorithms can be categorized along different dimensions based either on the underlying methodology of the algorithm, leading to partition or agglomerative approaches; the structure of the final solution, leading to hierarchical or nonhierarchical solutions; the characteristics of the space in which they operate, leading to feature or similarity approaches.

2.1. Agglomerative And Partitional Algorithms

Partitional algorithms, such as *K*-means [19], *K*-medoids [16], probabilistic [6], graph partitioning based [13], or spectral based [16], find the clusters by partitioning the entire dataset into either a predetermined or an automatically derived number of clusters. Partitional clustering algorithms compute a *k*-way clustering of a set of objects either directly or through a sequence of repeated bisections. A direct *k*-way clustering is commonly computed as follows. Initially, a set of *k* objects is selected from the datasets to act as the seeds of the *k* clusters. Then, for each object, its similarity to these *k* seeds is computed, and it is assigned to the cluster corresponding to its most similar seed. This forms the initial *k*-way clustering. This clustering is then repeatedly refined by recalculating the new seed so that it optimizes a desired clustering criterion function. A k-way partitioning through repeated bisections is obtained by recursively applying the above algorithm to compute two-way clustering (i.e., bisections). Initially, the objects are partitioned into two clusters, and then one of these clusters is selected and is further bisected, and so on.

This process continues k–1 times, leading to k clusters. Each of these bisections is performed so that the resulting two way clustering solution optimizes a particular criterion function. Criterion functions used in the partitional clustering reflect the underlying definition of the "goodness" of clusters. The partitional clustering can be considered as an optimization procedure that tries to create high-quality clusters according to a particular criterion function. Many criterion functions have been proposed [28]. Criterion functions measure various aspects of intracluster similarity, intercluster dissimilarity, and their combinations. These criterion functions use different views of the underlying collection, by either modeling the objects as vectors in a high-dimensional space or by modeling the collection as a graph. Hierarchical agglomerative algorithms find the clusters by initially assigning each object to its own cluster and then repeatedly merging pairs of clusters until a certain stopping criterion is met. The three basic criteria to determine which pair of clusters to be merged next are single-link, complete-link, and group average (UPGMA - unweighted pair group method with arithmetic mean) [16]. The single-link criterion function measures the similarity of two clusters by the maximum similarity between any pair of objects from each cluster, whereas the complete-link criterion uses the minimum similarity. In general, both the single-link and the complete-link approaches do not work very well because they either base their decisions to a limited amount of information (single-link) or assume that all the objects in the cluster are very similar to each other (complete link). On the other hand, the group average approach measures the similarity of two clusters by the average of the pair wise similarity of the objects from each cluster and does not suffer from the problems arising with single-link and complete link .In addition to these three basic approaches, a number of more sophisticated schemes have been developed, such as CURE [22], ROCK [23], and CHAMELEON [18] that has been shown to produce superior results. CURE [22] is more robust to outliers, and identifies clusters having non-spherical shapes and wide variances in size. ROCK [23] is a robust hierarchical clustering algorithm for categorical attributes that employs links and not distances when merging clusters and provides good scalability. CURE and ROCK are designed to find clusters that fit some static models. These algorithms can breakdown if the choice of parameters in the static model is incorrect with respect to the data set being clustered. But CHAMELEON [18] measures the similarity of two clusters based on a dynamic model. In the clustering process, two clusters are merged only if the inter-connectivity and closeness (proximity) between two clusters are high relative to the internal inter-connectivity of the clusters and closeness of items within the clusters. The merging process using the dynamic model presented facilitates discovery of natural and homogeneous clusters.Finally, hierarchical algorithms produce a clustering solution that forms a dendogram, with a single all-inclusive cluster at the top and single-point clusters at the leaves. In contrast, in nonhierarchical algorithms there tends to be no relation between the clustering solutions produced at different levels of granularity.

3. Similarity Measures

In most microarray clustering applications the goal is to find clusters of genes or clusters of conditions. A number of different methods have been proposed for computing these similarities, including Euclidean distance-based similarities, correlation coefficients, and mutual information. The use of correlation coefficient-based similarities is primarily motivated by the fact that while clustering gene expression datasets, the expression levels

of different genes are related under various conditions. The correlation coefficient values between genes is estimated by the Pearson correlation coefficient, which is given by

$$\text{sim}(v,u) = \text{corr}(v,u) = \frac{\sum_{i=1}^{n}(v_i - \bar{v})(u_i - \bar{u})}{\sqrt{\sum_{i=1}^{n}(v_i - \bar{v})^2}\sqrt{\sum_{i=1}^{n}(u_i - \bar{u})^2}},$$

and it can be used directly or transformed to absolute values if genes of both positive and negative correlations are important in the application. An alternate way of measuring the similarity is to use the mutual information between a pair of genes. The mutual information between two information sources A and B represent how much information the two sources contain for each other. D'Haeseleer et al [8] used mutual information to define the relationship between two conditions A and B. A feature common to many similarity measures used for microarray data is that they almost never consider the length of the corresponding gene or condition vectors, which is the actual value of the differential expression level, but focus only on various measures of relative change or how these relative measures are correlated between two genes or conditions [29]. The reason for this is twofold. First, there are still significant experimental errors in measuring the expression level of a gene, and is not reliable to use it "as is." Second, in most cases interest is shown on how the different genes change across the different conditions (i.e., either upregulated or downregulated) and the interest is not shown in the exact amount of this change.

4. Limitations Of The Conventional Clustering Schemas

Since the early days of the development of the micro array technologies, a wide range of existing clustering algorithms have been used, and novel new approaches have been developed for clustering gene expression datasets. The most effective traditional clustering algorithms are based either on the group-average variation of the agglomerative clustering methodology, or the K-means approach applied to unit-length gene or condition expression vectors. Agglomerative solutions are inherently suboptimal when compared to partitional approaches, which allow for a wider range of feasible solutions at various levels of cluster granularity. However, despite this, the agglomerative solutions tend to produce reasonable and biologically meaningful results, and allow for an easy visualization of the relationships between the various genes or conditions in the experiments. The ease of visualizing the results has also led to the extensive use of self-organizing maps (SOM) for gene expression clustering [25]. However, as the dimensionality of these datasets continues to increase (primarily by increasing the number of conditions that are analyzed), requiring consistency across the entire set of conditions will be unrealistic. These algorithms tend to produce local solutions and hence genetic algorithms are integrated to provide good heuristic solutions.

5. Dimensionality Reduction

Many new algorithms have been proposed recently to tackle the problem of clustering gene expression data with high dimensionality. The basic idea of global dimension reduction is to compress the entire gene/ condition matrix to represent genes by vectors in a compressed space of low dimensionality, such that the biologically interesting results can be extracted [20].Alter et al [2] proposed to use singular value decomposition (SVD) to compress the data and then apply traditional clustering algorithms (such as k-means). They also showed that their algorithms can find meaningful clusters on cancer cells, leukemia dataset and yeast cell cycle

dataset. Finding clusters in subspaces tackle this problem differently by redefining the problem of clustering as finding clusters whose internal similarities become apparent in subspaces or clusters that preserve certain expression patterns among the dimensions in subspaces [13]. The various algorithms differ from one another in how they model the desired clusters, the optimization algorithm and clustering algorithm that generate the desired clusters, and whether the algorithms allow genes that belong to more than one cluster (i.e., overlapping clusters). Cheng and Church [7] assume each expression value in the matrix as the addition of three components: the background level, the row effect, and the column effect. Thus, they use minimum mean squared residue as the objective function to find clusters in subspaces that have small deviations with respect to the rows in the cluster, the columns in the subspace, and the background defined by the cluster. Correlation clustering groups the data sets into subsets called correlation clusters such that the objects in the same correlation cluster are all associated to a common hyperplane of arbitrary dimensionality. The prominent application for correlation clustering is the analysis of gene expression data. Gene expression data contain the expression levels of thousands of genes, indicating how active the genes are, according to a set of samples. A common task is to find clusters of co-regulated genes, i.e. clusters of genes that share a common linear dependency within a set of their features. The first approach that can detect correlation clusters is ORCLUS [1] that integrates PCA into *k*-means clustering. The algorithm 4C [5] integrates PCA into a density-based clustering algorithm. Elke Achtert [10] proposed correlation clustering algorithm COPAC (Correlation PArtition Clustering) that aims at improved robustness, completeness, usability, and efficiency.

6. Genetical Gorithms (GA)

In GAs, the search space of a problem is represented as a collection of individuals [11]. The individuals are represented by character strings, which are referred to as *chromosomes*. A collection of such strings is called the *population*. The purpose is to find the individual from search space with the best genetic material. The quality of an individual is measured with an objective function or the fitness function. Based on the principle of survival of the fittest, a few of the strings are selected and each is assigned to a number of copies that go into the mating pool. Biologically inspired operators like *crossover* and *mutation* are applied on these strings to yield a new generation of strings. The process of *selection*, *crossover* and *mutation* continues for a fixed number of generations or till the termination condition is satisfied.

7. Criteria For Evaluating Clustering Algorithms

Considering the characters of gene expression data, and the particular applications in functional genomics, the optimal algorithms for analysis of gene expression data need the following properties suggested by Jain et al and Han et al [17,14]

- Scalability and efficiency: Algorithms should be efficient and scalable considering the large amount of data to be handled.
- Irregular shape: Algorithms need to be able to identify a dense set of points which forms a cloud of irregular non spherical shapes.
- Robustness: The clustering mechanisms should be robust against large amounts of noise and outlier.
- Order insensitivity: Algorithms should not be sensitive to the order of input. That is, clustering results should be independent of data order.

- Number of Clusters: The number of clusters inside the data set needs to be determined by the algorithm itself and not prescribed by the user.
- Parameter estimation: The algorithms should have the ability to estimate any parameters required by the algorithm from the data set, and no domain knowledge input is required from the user.
- Dimensionality: Algorithms need the ability to handle data with high dimensionality or the ability to find clusters in subspaces of the original space.
- Stability: No data object will be classified into different clusters for different running of the algorithm.
- Incrementability: Algorithms should be able to incrementally handle the addition of new data or the deletion of old data instead of re-running the algorithms on the new data set.
- Interpretability: The clustering results of the algorithms need to be interpretable. That is, clustering may need to be tied up with specific biological interpretations and applications.

8. Overview Of Microarray Technologies

DNA microarrays measure gene expression levels by exploiting the preferential binding of complementary, single-stranded nucleic acid sequences. cDNA microarrays, developed at Stanford University [Stanford University Genomic Resources : http ://genome-www.stanford.edu] are glass slides, to which single-stranded DNA molecules are attached at fixed locations (spots) by high-speed robotic printing . Each array may contain tens of thousands of spots, each of which corresponds to a single gene. mRNA from the sample and from control cells is extracted and cDNA is prepared by reverse transcription. Then, cDNA is labeled with two fluorescent dyes and washed over the microarray so that cDNA sequences from both populations hybridize to their complementary sequences in the spots. The amount of cDNA from both populations bound to a spot can be measured by the level of fluorescence emitted from each dye. For example, the sample cDNA is labeled with a red dye and the control cDNA is labeled with a green dye. Then, if the mRNA from the sample population is in abundance, the spot will be red; if the mRNA from the control population is in abundance, it will be green; if sample and control bind equal the spot will be yellow; if neither binds, it will appear black. Thus, the relative expression levels of the genes in the sample and control populations can be estimated from the fluorescent intensities and colors for each spot. After transforming the raw images produced by microarrays into relative fluorescent intensity with some image processing software, the gene expression levels are estimated as log-ratios of the relative intensities. A gene expression matrix can be formed by combining multiple microarray experiments of the same set of genes but under different conditions, where each row corresponds to a gene and each column corresponds to a condition (i.e. a microarray experiment) [27,3]. The Affymetrix GeneChip oligonucleotide array contains several thousand single-stranded DNA oligonucleotide probe pairs. Each probe pair consists of an element containing oligonucleotides that perfectly match the target (PM probe) and an element containing oligonucleotides with a single base mismatch (MM probe). A probe set consists of a set of probe pairs corresponding to a target gene. Similarly, the labeled RNA is extracted from sample cell and hybridizes to its complementary sequence. The expression level is measured by determining the difference between the PM and MM probes. Then, for each gene (i.e., probe set) average difference or log average can be calculated, where average difference is defined as the average difference

between the PM and MM of every probe pair in a probe set and log average is defined as the average log ratios of the PM/MM intensities for each probe pair in a probe set.

9. Genetic Algorithm Based Clustering For Microarray Data

9.1 Genclust

Genclust is a genetic algorithm for clustering gene expression data proposed by Vito Di Gesú et al [26]. It has two key features: (a) a novel coding of the search space that is simple, compact and easy to update (b) it can be used naturally in conjunction with data driven internal validation methods. It is experimented with the FOM methodology, specifically conceived for validating clusters of gene expression data. The validity of the algorithm has been assessed experimentally on real data sets, both with the use of validation measures and in comparison with other algorithms like Average Link, Cast, Click and K-means. GenClust is experimentally competitive with K-means, Click [21] and Cast [4]. Moreover, the algorithm is well suited for use in conjunction with data driven internal validation methodologies and in particular FOM, which has received great attention in the specialized literature. It defines clustering as an optimization problem Given a subset $Y = \{y1, y2, ..., ym\}$ of X, let $c(Y)$ denote the centroid of Y and let its variance be

$$VAR(Y) = \frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{d}(y_{i,j} - c(Y)_j)^2.$$

Given an integer k, we are interested in finding a partition of X into k classes $C0$, $C1$ ..., Ck-1 so that the total internal variance is

$$VAR(\mathcal{P}) = \sum_{i=0}^{k-1} VAR(C_i)$$

The algorithm proceeds in stages, producing a sequence of partitions, each consisting of k classes, until a halting condition is met. Following the evolutionary computational paradigm, a population evolves by means of genetic operators, i.e., cross-over, mutation and selection, resulting in a random walk in cluster space, where the fitness function gives a drift to the process towards a local optimum. Its performance is measured by adjusted Rand index and FOM. Table 1 [26] shows the adjusted Rand index and FOM values for the rat RCNS data set of various algorithms.

Table 1: Performance of the Algorithms for RCNS Rat

Method	AdjustedRand	FOM
GenClust random	0.168	3.89
Min kmeans-random	0.144	3.81
Max kmeans-random	0.258	3.81
Cast	0.12	3.98
Kmeans-Avlink	0.167	3.71
Avlink	0.19	4.05
GenClust-Avlink	0.161	4.07

The table shows that GenClust-AvLink is to be preferred to GenClust-Random. Moreover, GenClust- AvLink seems to take better advantage of the output of Average Link than K-means. It also appears that GenClust- AvLink is competitive with Average Link and K-means, Cast and Click.

9.2 Hgaclus

HGACLUS is suggested by Haiyan Pan et al [12]. In this paper ,the parallelism searching capability of GAs is used to design a clustering schema (HGA- CLUS) combining merits of the Simulated Annealing to find an optimal or near-optimal set of medoids whose size was predefined. According to this optimal set of medoids, each observation was allocated to the nearest medoid and the best k clusters were then constructed. Each string was evaluated using the following fitness function,

$$f(s_h) = \frac{\mathrm{traceB}/(k-1)}{\mathrm{traceW}/(n-k)}$$

where n and k are the total number of points and the number of clusters in the partition, respectively. B and W are the covariance matrices of between-cluster sums and the pooled within-cluster sums of squares, respectively.

$$p(s_h) = \frac{\exp(f(s_h)/T)}{\sum_{h=1}^{p} \exp(f(s_h)/T)}, \quad h = 1, 2, \ldots, p.$$

where T>0 is a cooling temperature. The cooling schedule function is

$$T(g) = \frac{G-g}{G} T_0, \quad g = 0, 1, \ldots, G-1.$$

The process of fitness computation, selection, crossover, and mutation was executed for G generations. Variance Ratio Criterion (VRC) and Silhouette Width are the validation indices used by the algorithm. With regard to the criteria of external isolation and internal consistency, HGACLUS appeared to perform as well as or better than any of the other mentioned methods for multi-class clustering. The fig 1 shows [12] the results for the datasets.

9.3 Hybrid Method Using EM Algorithm

Zeke et al [32] suggest the hybrid method using EM algorithm. In this paper a framework is proposed that hybridizes Evolutionary Computation, in particular Genetic Algorithm with a local-learning algorithm to perform optimal clustering of time-course gene expression data. The hybrid algorithm combines the strengths of GA and the local-learning algorithm (EM) by using the former to select subset of data as initial cluster centers and the later to perform fast local optimization to achieve the final centers from these initial centers. In this way, the optimality of the final centers returned by the local-search algorithm can be used as the objective function for GA, which searches for the globally optimal subset of data as initial cluster centers. In other words, rather than beginning the local optimization from data points that are randomly chosen, the hybrid algorithm begins the local optimization from data points that are globally optimal, therefore increasing the consistency of the final clustering solution. The hybrid algorithm is applied to

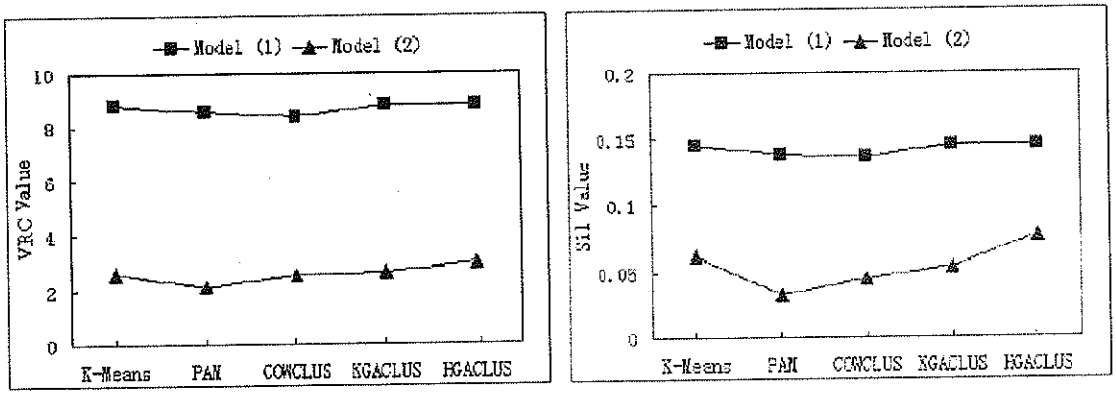

Figure 1 : The average VRC and Silhouette Width Values of Five Clustering Methods for Models 1(3 sets of cancer genes) and Model 2 (5 sets of cancer genes)

the human fibroblasts time course data that requires clustering of 512 useful genes and shows superior performance over using EM alone. Results show that although the hybrid algorithm requires higher computational cost, it performs consistently better in clustering accuracies. The hybrid framework is applied

to gene clustering with a mixture of Multiple Linear Regression models (MLRs), which uses Expectation Maximization algorithm (EM) as the local learning algorithm. The fitness function of each solution is measured as the maximum log likelihood of the EM-optimized model. For selection etilist scheme is used. The clustering model is a mixture of G MLRs (one for each cluster), each of which represents a single gene trajectory cluster given by

$Y_i = S(\mu_k + \gamma_i) + \varepsilon_i \quad \gamma_i \sim N(0,\Gamma) \quad \varepsilon_i \sim N(0,R)$ where $Yi = [y_{i1}, y_{i2}, \ldots y_{il}]t$ is the ith gene trajectory of length l. The log normalized data of 517 genes is shown

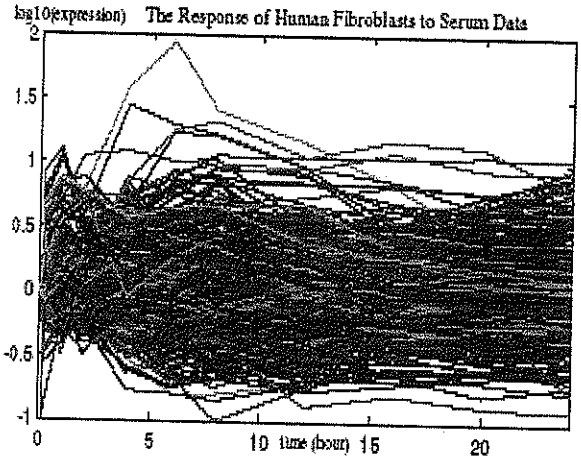

Figure 2 : The Human Fibroblasts To Serum Data (517 Genes).

A typical GA run is shown in Fig. 3 [32]. The fitness increases uni-directionally, which is the characteristic of the etilist selection scheme.

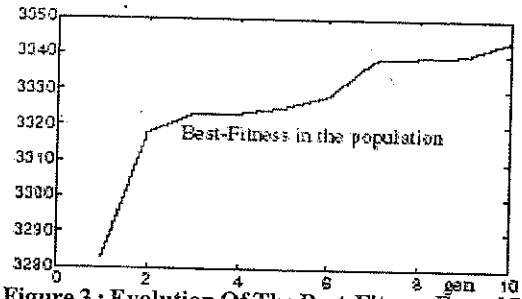

Figure 3 : Evolution Of The Best-Fitness Found In The Population.

The experimental results on gene expression time course data available on the public domain show the advantages of the hybrid GA-EM approach when compared with the standard approach of using random initialization EM algorithm only.

9.4 Multi Objective Genetic Clustering Algorithm

Sanghamitra Bandyopadhyay, Anirban Mukhopadhyay and Ujjwal Maulik [24] proposed a two-stage clustering algorithm, that employs a recently proposed variable string length genetic scheme and a multi objective genetic clustering algorithm. It is based on the novel concept of points having significant membership to multiple classes. An iterated version of the well known Fuzzy C-Means is also utilized for clustering. The significant superiority of the proposed two-stage clustering algorithm as compared to the average linkage method, Self Organizing Map (SOM) and a recently developed weighted Chinese restaurant based clustering method (CRC), widely used methods for clustering gene expression data, is established on a variety of artificial and publicly available real life data sets. The biological relevance of the clustering solutions is also analyzed.

The Table 2 shows [24] maximum selectivity of different clusters produced by different algorithms on Yeast Sporulation data. The following Table 3 shows the comparison of various GA based methods used for gene data clustering.

Table 2: Maximum Selectivity Of Different Clusters Produced By Different Algorithms On Yeast Sporulation Data

Algorithms	Maximum selectivity of clusters (%)							
	C-1	C-2	C-3	C-4	C-5	C-6	C-7	C-8
SiMM-TS	61.36	29.33	46.38	31.43	58.39	30.84	-	-
VGA	20.70	15.94	17.97	27.41	43.67	50.42	-	-
IFCM	28.53	54.45	10.08	14.86	23.73	34.50	21.60	-
Avg. Link	3.91	-	5.36	10.95	4.57	-	-	-
SOM	16.26	50.34	34.50	18.11	22.69	15.73	-	-
CRC	21.60	22.32	32.04	56.55	29.68	46.29	22.68	20.00

Table 3: Comparison Of GA Based Clustering Algorithms

Method	Approach	Validation Indices	Data set
1. GenClust	GA based partitional clustering	FOM & Rand index	Rat RCNS
2. HGAclust	GA based simulated annealing concept	VRC & silhouette width	Embryonal CNS data
3. Hybrid EM	GA based on mixture of MLR's	Maximum log likelihood / BIC	Human fibroblast to serum data
4. Multi objective clustering	GA based Fuzzy C means	ARC & silhouette width	Yeast Sporulation data

10. Conclusion

In this survey several genetic algorithm based clustering techniques for micro array gene data are presented. Some preliminary concepts that form the basis for the development of clustering algorithms are discussed. This paper provides a framework for the evaluation of clustering in gene expression analysis.

References

1. Aggarwal C C and Yu P S, " *Finding generalized projected clusters in highdimensional spaces"* , In Proc.ACM SIGMOD,PP.70-81, 2000.

2. Alter P O Brown and Bostein D , "*Singular value decomposition for genome-wide expression data processing and modeling*", Proceedings of Natural Academy Sciences , USA, Vol. 97 , No. 18, PP.10101-10106, 2000.

3. Baldi P and Long A D, "*A Bayesian framework for the analysis of microarray expression data: regularized t -test and statistical inferences of gene changes*", Bioinformatics, Vol 17, PP.509-519, 2001.

4. Ben-Dor A, Shamir R, Yakhini Z, " *Clustering of gene expression patterns*", Journal of Computational Biology, Vol. 6, PP. 281-297, 1999.

5. Bohm C, Kailing K, Kroger P and Zimek A, " *Computing clusters of correlation connected objects",* In Proc. ACM SIGMOD , France, PP. 455-467, 2004.

6. Cheeseman P and Stutz J, " *Baysian classification (autoclass):Theory and results*", Advances in Knowledge Discovery and Data Mining, AAAI/MIT Press , PP. 153–180, 1996.

7. Cheng Y and Church G M, "*Biclustering of expression data*", ISMB'00, Vol. 8, PP.93-103, 2000.

8. D'haeseleer P, Fuhrman S , Wen X and Somogyi . R, "*Mining the gene expression matrix: Inferring gene relationships from large scale gene expression data*", Information Processing in Cells and Tissues, Plenum Publishing,PP. 203–212, 1998.

9. Eisen M B, Spellman P T, Brown P O, Botstein D, " Cluster analysis and display of genome-wide expression patterns", *Proc Natl Acad Sci, U S A. , 95* , PP. 14863–14868 , doi: 10.1073/pnas.95.25.14863, 1998.

10 .Elke Achtert et al , "*Robust, Complete, and Efficient Correlation Clustering*", In Proc. SIAM Int. Conf. on Data Mining (SDM), Minneapolis, MN,PP.413-418, 2007.

11. Goldberg D, "*Genetic Algorithms in Search, Optimization and Machine Learning Reading*", MA, Addison Wesley, 1989.

12. Haiyan Pan, Jun Zhu1 and Danfu Han, *"Genetic Algorithms Applied to Multi-Class Clustering for Gene Expression Data*", Geno., Prot. & Bioinfo. , Vol. 1 ,No. 4, PP.279-287, 2003.

13. Han E H, Karypis G, Kumar V and Mobasher B, " *Hypergraph based clustering in high-dimensional data sets: A summary of results*", IEEE Bulletin of the Technical Committee on Data Engineering, Vol. 21, No .1,PP.15-22, 1998.

14. Han J and Kamber M , "*Data Mining: Concepts and Techniques*", The Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann Publishers, 2000.

15. Han J, Kamber M and Tung A K H, *" Spatial clustering methods in data mining: A survey*", Geographic Data Mining and Knowledge Discovery, Taylor and Francis, 2001.

16.Jain A K and Dubes R C, " *Algorithms for Clustering Data"*, Prentice Hall, 1988.

17. Jain A K, Murty M N, Flynn P J, " *Data clustering: A review*", ACM Computing Surveys, Vol. 31 , No 3, PP.264–323, 1999.

18. Karypis G, Han E H, Kumar V, "*Chameleon: A hierarchical clustering algorithm using dynamic modeling*", IEEE Computer, Vol. 32, No 8, PP. 68–75, 1999.

19.MacQueen J, "*Some methods for classification and analysis of multivariate observations*", In Proceedings of the 5th Symposium on Mathematical Statistics and Probability , PP. 281–297, 1967.

20. Raychaudhuri S, Stuart J M and Altman R B, "*Principal components analysis to summarize microarray experiments: application to sporulation time series*", In Pacific Symposium on Biocomputing, PP. 415-426, 2000.

21.Sharan R, Maron-Katz A, Shamir R, "*CLICK and EXPANDER: a system for clustering and visualizing gene expression data*", Bioinformatics , 19 ,PP.1787-1799,2003.

22. Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim, " *CURE: An efficient clustering algorithm for large databases*", In Proceedings of ACM-SIGMOD International Conference on Management of Data, PP.73-84, 1998.

23. Sudipto Guha, Rajeev Rastogi and Kyuseok Shim, "*ROCK: a robust clustering algorithm for categorical attributes*", In Proceedings of the 15th International Conference on Data Engineering,PP.345-366, 1999.

24. Sanghamitra Bandyopadhyay, Anirban Mukhopadhyay and Ujjwal Maulik, "*An improved algorithm for clustering gene expression data*", Bioinformatics, Vol.23,PP. 2859-2865, 2007.

25. Tamayo P, Slonim D, Mesirov J, Zhu Q, Kitareewan S, Dmitrovsky E and Lander E S and Golub T R, "*Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation*", Proceedings of Natural Academy Science, Vol. 96 , PP.2907–2912, 1999.

26. Vito Di Gesú, "*GenClust: A genetic algorithm for clustering gene expression data*", BMC Bioinformatics , Vol.6 , doi:10.1186/1471-2105-6-289, 2005

27. Yang Y H , Dudoit S, Luu P, and Speed T P, "*Normalization for cdna microarray data*", In SPIE International Biomedical Optics Symposium,2001,citeseer.ist.psu.edu/406329.html

28. Ying Zhao and George Karypis, "*Criterion functions for document clustering: Experiments and analysis*", Technical Report TR #01–40, Department of Computer Science, University of Minnesota, Minneapolis, 2001, Available online *http://cs.umn.edu/~karypis/publications*.

29. Yeung K Y , Haynor D R, and Ruzzo W L, "*Validating clustering for gene expression data*", Bioinformatics, Vol. 17, No 4, PP.309–318, 2001.

30. Yona G, Linial N, and Linial Protomap M, "*Automatic classification of protein sequences and hierarchy of protein families*", Nucleic Acids Research, 28 , PP.49–55, 2000.

31. Zhao Y and Karypis G, "*Comparison of agglomerative and partitional document clustering algorithms*", In *SIAM (2002) workshop on Clustering High-dimensional Data and Its Applications*, available as technical report #02-014 university of Minnesota, 2002.

32. Zeke S H Chan and Nikola Kasabov, "*Gene Trajectory Clustering with a Hybrid Genetic Algorithm and Expectation Maximization Method*", Journal of Bioinformatics and Computational Biology, Vol. 3 , No 5, PP. 1227-1241, 2005 .

Author's Biography

Dr. K. Vivekanandan received the Ph.D. degree in Computer science from Bharathiar University, India. He is currently Reader in Bharathiar University, India. He has a total teaching experience of 21 years. He has published 17 papers in international and national journals .He has produced 5 Ph D's and 11 MPhil scholars in computer science. His research interests include data mining and knowledge discovery, and Management Information System. .

P.Krishnakumari received the MPhil degree in Computer science from Bharathiar University, India. She is currently pursuing her Ph.D in computer science and has a total 13 years of teaching experience. At present she is a senior lecturer in the Department of computer science of Sri Ramakrishna college of Arts and Science for women, Coimbatore. She has presented research papers in international and national conferences and published papers in international journals. Her research interests include data mining and knowledge discovery, genetic algorithms, Bioinformatics, image compression, networking.