SA-Kmeans: A Novel Data-Mining Approach to Identifying and Validating Gene Expression Data

Santanu Kumar Rath[1] and Kumar Dhiraj [2]

Abstract--**Data Mining has become an important topic in effective analysis of gene expression data due to its wide application in the biomedical industry. In this paper a novel simulated annealing (SA) based k-means clustering cum validation technique has been implemented. SA is a random-search technique which exploits an analogy between the way in which a metal cools and freezes into a minimum energy crystalline structure and the search for an optimal solution. We have implemented proposed algorithm and tested their performance using two real biological data of cancer of [118x60] samples and breast-cancer data of [700x9] samples. The results of proposed algorithm were Compared with the k-means algorithm with very comparable results. This approach to gene expression analysis that integrates SA, k-means and cluster validation techniques simultaneously results in a robust, high quality and highly efficient algorithm.**

Index Terms--**Bio-informatics, Cancer-Genomics, Gene-expression, Data-mining, Cluster validation, k-mean, Simulated Annealing, Microarray.**

I. INTRODUCTION

SIMULATED annealing (SA),[12,13,14,15,16] in a more general system; it forms the basis of an optimization technique for combinatorial and other problems. By analogy with this physical process, each step of the SA algorithm replaces the current solution by a random "nearby" solution, chosen with a probability that depends on the difference between the corresponding function values and on a global parameter T (called the temperature), that is gradually decreased during the process. The dependency is such that the current solution changes almost randomly when T is large, but increasingly downhill" as T goes to zero. The allowance for "uphill" moves saves the method from becoming stuck at local minima—which are the bane of greedier methods.

In recent years, the DNA microarray [5] has become an important and widely used technology since it enables the possibility of examining the expressions of thousands of genes simultaneously in a single experiment. A key step in the analysis of gene expression data is the detection of gene groups that manifest similar expression patterns. The main algorithmic problem here is to cluster multi-conditions gene expression patterns. Basically, a cluster algorithm partitions entities into groups based on the given features of the entities, so that the clusters are homogeneous and well separated. A variety of clustering methods have been proposed for the mining of gene expression data [2, 3, 6].Although a number of clustering methods have been studied in the literature, they are not satisfactory in terms of: 1) quality, and 2) efficiency. In this paper, we propose an integrated approach to identifying and validating clusters in multivariate datasets and apply it to the mining of multi-conditions gene expression data. This approach iterative computing process is adopted to meet the requirement of efficiency. Through experiments conducted on real gene expression data, the proposed approach is shown to deliver higher efficiency, and clustering quality than other methods.

The rest of the paper is organized as follows: In section II, some related studies are introduced. Our approach is described in section III. Experiments conducted to evaluate the performance of the proposed method are presented in section IV. Conclusions and future works are given in section V. Section VI, VII and VII contains Acknowledgement, References and Biography respectively.

II. RELATED WORK

In recent years, a number of clustering methods have been proposed, and they can be classified into several different types: partitioning-based methods (e.g., k-means[10], k-medoids, PAM, and CLARA) [9], hierarchical methods (e.g., UPGMA [10], CURE [8]), density-based methods (e.g., CAST [3], DBSCAN [7]), grid-based methods (e.g., CLIQUE [1]), model-based methods (e.g., SOM [11]), etc. Among them, several methods have been applied to cluster gene expression datasets, such as in [2, 3, 6].

Although a number of clustering algorithms have been proposed, they may not find the best clustering result efficiently based on the given dataset. An important problem involved here is how to validate the clustering result. The main drawback of the existing clustering methods when applied for gene expression pattern analysis is that they can not meet the requirements of

high quality and high efficiency simultaneously during the analysis process. In the following, we describe a new approach to gene expression analysis that integrates clustering and validation techniques in such a way that high quality and high efficiency are achieved simultaneously.

Santanu Kumar Rath AND Kumar Dhiraj are with Department of Computer Science & Engg. at National Institute of Technology, Rourkela, Orissa, INDIA.
Email id: rath.santanu@gmail.com1,
kumardhiraj.nit.rourkela@gmail.com2.
Phone: 91-661- 2462357.
Fax: 91-661- 2462999, 2472926.

III. PROPOSED APPROACH

In this section, we first define the problem. Then we describe our approach in detail, including the principles behind it.

A. Problem Definition

The problem of multivariate gene expression clustering can be described briefly as follows. Given a set of genes with unique identifiers, a vector $E_i = \{E_{i1}, E_{i2} \ldots E_{in}\}$ is associated with each gene i, where E_{ij} represents the response of gene i under condition j. The goal of gene expression clustering is to group genes based on similar expressions over all the conditions. That is, genes with similar corresponding vectors should be classified into the same cluster.

B. Proposed Method

The main steps in the proposed approach are shown in Fig.1. Given a piece of gene expression data, the **Step one** is Data preprocessing. In general it can be done by simple transformations or normalizations performed on single variables, filters, calculation of new variables from existing ones. In our propose work, only the first of these is implemented. Scaling of variables is of special importance, since we have used Euclidean metric to measure distances between vectors. If one variable has values in the range of [0,...,1000] and another in the range of [0,...,1] the former will almost completely dominate the cluster formation because of its greater impact on the distances measured. Typically, one would want the variables to be equally important. The standard way to achieve this is to linearly scale all variables so that their variances are equal to one.

Step two is to diversify the data using simulated annealing method. It has been proved that by carefully controlling the rate of cooling of the temperature, SA can find the global optimum. SA's major advantage over other methods is an ability to avoid becoming trapped in local minima. The Algorithm employs a random search which not only accepts changes that decrease the objective function f (assuming a minimization problem), but also some changes that increase it. The latter are accepted with a probability

$$p = \exp(-df / T) \qquad (1)$$

where df is the increase in f and T is a control parameter, which by analogy with the original application is known as the system "temperature" irrespective of the objective function involved. The implementation of the basic SA algorithm is shown in Fig. 2.

Step three: Begin with a decision on the value of k = number of clusters

Step four: Put any initial partition that classifies the data into k clusters. Assign the training samples randomly, or systematically as the following:

Take the first k training sample as single-element clusters

Assign each of the remaining (N-k) training sample to the cluster with the nearest centroid. After each assignment, recomputed the centroid of the gaining cluster.

Step five: Take each sample in sequence and compute its distance from the centroid of each of the clusters. If a sample is not currently in the cluster with the closest centroid, switch this sample to that cluster and update the centroid of the cluster gaining the new sample and the cluster losing the sample.

Step six: Repeat step 5 until convergence is achieved, that is until a pass through the training sample causes no new assignments.

If the number of data is less than the number of cluster then we assign each data as the centroid of the cluster. Each centroid will have a cluster number. If the number of data is bigger than the number of cluster, for each data, we calculate the distance to all centroid and get the minimum distance. This data is said belong to the cluster that has minimum distance from this data.

Since we are not sure about the location of the centroid, we need to adjust the centroid location based on the current updated data. Then we assign all the data to this new centroid. This process is repeated until no data is moving to another cluster anymore. Mathematically this loop can be proved to be convergent. The convergence will always occur if the following condition satisfied:

Each switch in step 4 the sum of distances from each training sample to that training sample's group centroid is decreased.

There are only finitely many partitions of the training examples into k clusters.

In the final step, a validation test is performed to evaluate the quality of the clustering result produced in step fourth.

A. Cluster Validation

There are various form of definition are available in literature to find out the homogeneity and separation of cluster. For example,

$$H_{ave} = \frac{1}{N} \sum_{Ci \in C} \| Ci \| \bullet H_2(Ci)$$

$$S_{ave} = \frac{1}{\sum_{ci \in cj} ||ci|| \bullet ||cj||} \sum_{ci \neq cj} (||ci|| \bullet ||cj||) S_2(ci, cj)$$

TABLE I
HOMOGENEITY VS. SEPARATION VALUES FOR K-MEANS AND PROPOSED ALGORITHM.

Datasets	Data Size	K-means Algorithm			SA-kmeans Algorithm		
		No. of clusters	H	S	No. of clusters	H	S
Datasets I	[10x3]	3	0.319	0.1173	3	0.2374	0.0711
Datasets II	[118x60]	8	0.463	0.894	8	0.2696	0.2768
		10	0.3582	0.8749	10	0.2043	0.3771
		15	0.2034	0.7829	15	0.1185	0.3086
		20	0.1372	0.6326	20	0.0766	0.2930
Datasets III	[699x9]	15	0.4659	0.5478	15	0.3020	0.3518
		20	0.2908	0.6272	20	0.2033	0.3131
		30	0.1526	0.5524	30	0.1156	0.316
		40	0.0931	0.5901	40	0.0765	0.3403

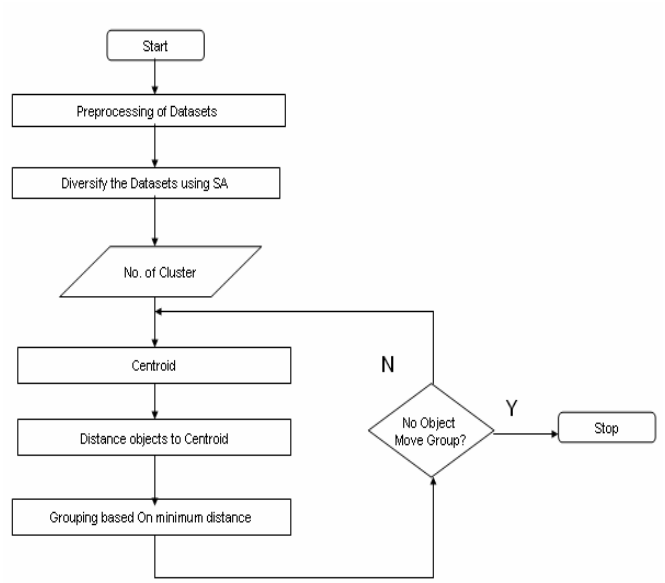

Fig. 1: Flow-Chart for SA-kmeans Algorithm.

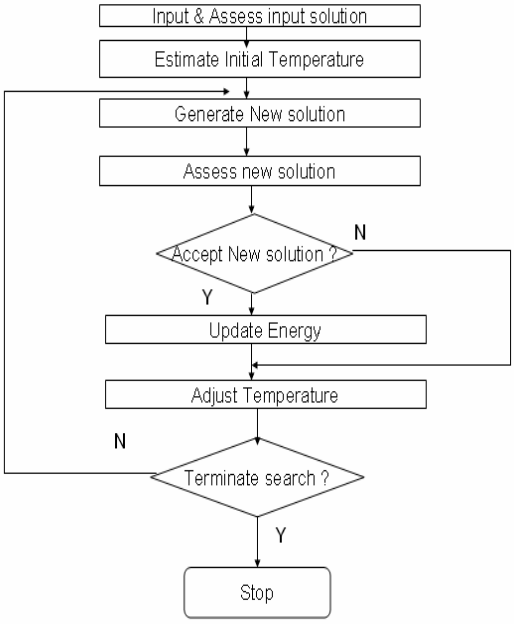

Fig. 2: Flow-Chart for diversification of Dataset using SA.

TABLE II
PARAMETER USED FOR SIMULATED ANNEALING.

No. of iteration	1000
Stopping criterion	99.9
Geometrical cooling(α)	0.9
Initial Temperature	1
Learning rate(l)	1

IV. EXPERIMENTAL EVALUATION

To validate the feasibility and performance of the propose approach, we implemented the approach in MATLAB 7.0(Pentium-4 CPU, 2.40GHz, 256 RAM) and applied it to both of real gene expression data and synthetic data.

A. Experimental Setup

To evaluate the performance of our approach, we used two real and one synthetic gene expression data.
Datasets I is a synthetic data of [10x3] Matrix.

Datasets II is a real biological data of [118x60] Matrix that represents growth inhibition factors when 118 drugs with putatively understood methods of action were applied to the NCI60 cell lines. The original data can be downloaded from http://discover.nci.nih.gov/nature2000/data/selected_data/a_matrix118.txt.

Datasets III represents wisconsin breast cancer Datasets and is obtained from the University Medical Centre, Institute of Oncology, Ljubljana, Yugoslavia (http://mlearn.ics.uci.edu or ftp://ftp.ics.uci.edu/pub/machine-learning-databases/breast- cancer-wisconsin). The dataset contained the expressions of 700 genes under 9 experimental

conditions. Attributes 2 through 9 have been used to represent instances (genes).
Each instance (genes) has one of two possible classes: benign or malignant.
Table II summarizes the parameter used for SA.

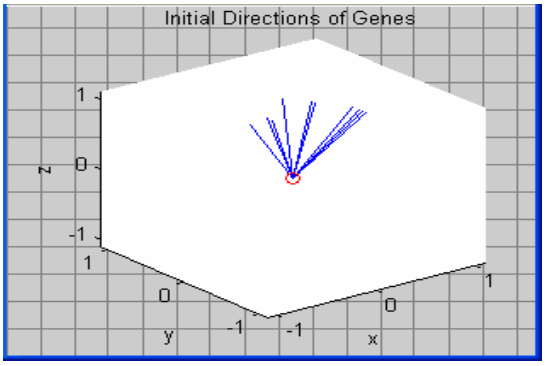

Fig. 3: Initial Unit Directions for Datasets I

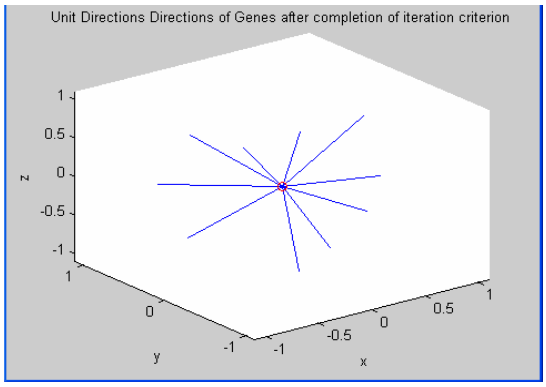

Fig. 4: Final Unit Directions for Datasets I

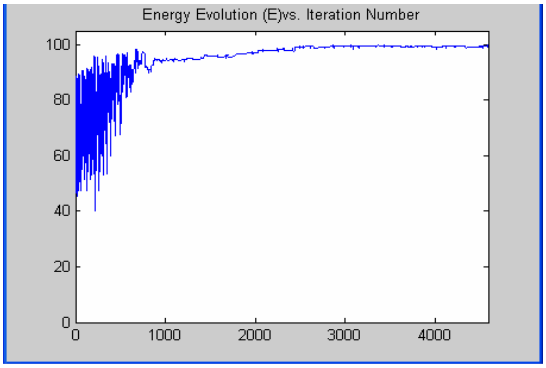

Fig. 5: Energy evolution for Datasets I

B. Results and Discussion

We have implemented and compared our approach with the well-known clustering method, namely, K-means [10]. The quality of the clusters formed was assessed using Homogeneity vs. separation values.

Fig. 3, 4, 5 shows result of SA on Datasets I.
Fig. 6, 7, 8 shows result of SA on Datasets II.
Fig. 9, 10, 11 shows result of SA on Datasets III.
Fig. 3, 6, 9 represents the unit direction of genes before implementing SA on Datasets I, Datasets II and Datasets III respectively.
Fig. 4, 7, 10 represents the unit direction of genes after implementing SA on Datasets I, Datasets II and Datasets III respectively.
These diagrams show that gene were diversified and henceforth results in good cluster formation.
Fig. 5, 8, 11 represents the Energy Evolution of genes Datasets I, Datasets II and Datasets III.
These figure show that after 1,000 iteration change in Energy almost becomes constant and this tells that gene were completely diversified.
The experimental results for Datasets I, II and III have been shown in Table I.
Table I summarizes the Homogeneity vs. Separation value for both k-means and Propose algorithm.
Fig. 12 represents the Homogeneity & Separation value for Cancer Datasets whereas Fig. 13 for Breast Cancer Datasets. Higher the Homogeneity value (Intra-cluster distance within clusters) and lower the Separation value (Intera-cluster distance between clusters) represents good cluster formation.

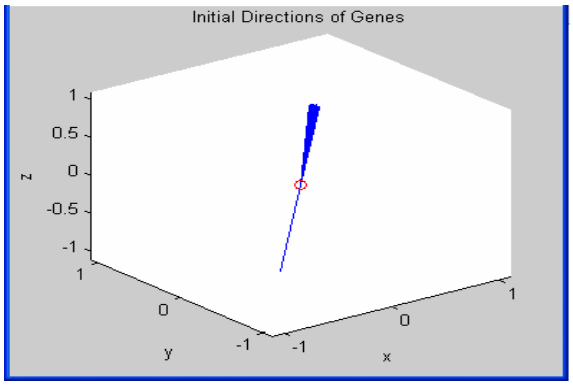

Fig. 6: Initial Unit Directions for Datasets II

Fig. 12 and Fig. 13 shows that Propose Algorithm performs better than K-means algorithm and gives good clustering results. As far as Homogeneity is concerned, k-means gives better result to Sa-kmeans, but for separation value Sa-kmeans gives better result than k-means. If we consider Homogeneity and Separation value together, SA-kmeans outperforms kmeans. It is also important to note that SA-kmean gives comparable value of Homogenity and Separation and that shows that Proposed algorithm is more robust in nature than kmean.

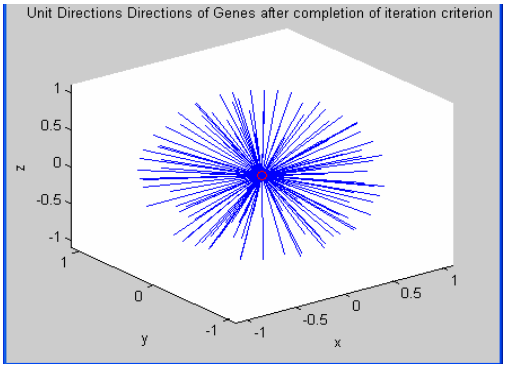

Fig. 7: Final Unit Directions for Datasets II

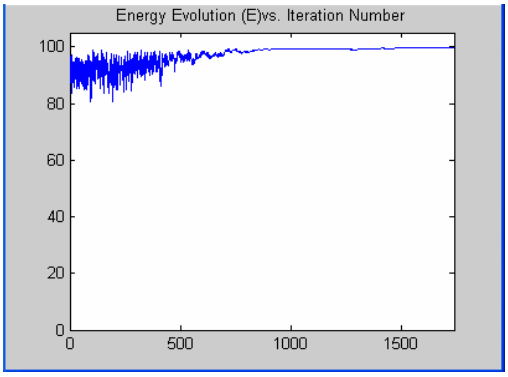

Fig. 8: Energy evolution for Datasets II

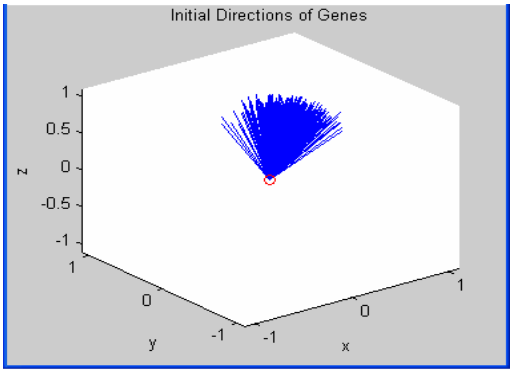

Fig. 9: Initial Unit Directions for Datasets III

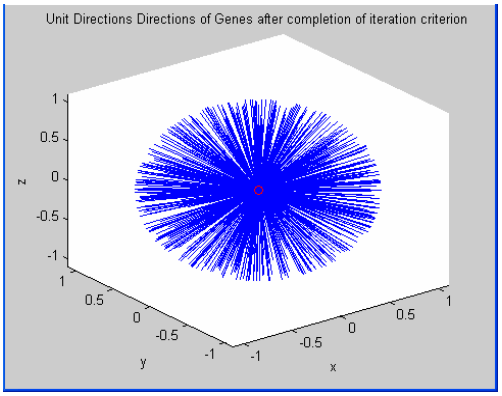

Fig. 10: Final Unit Directions for Datasets III

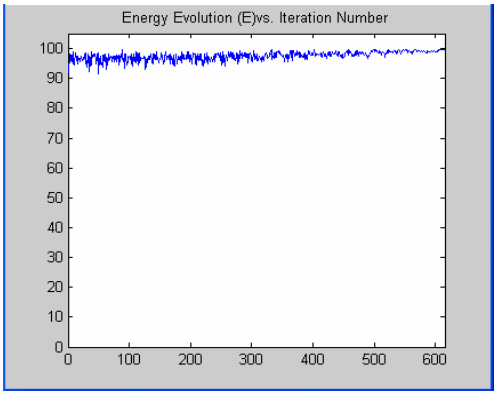

Fig. 11: Energy evolution for Datasets III

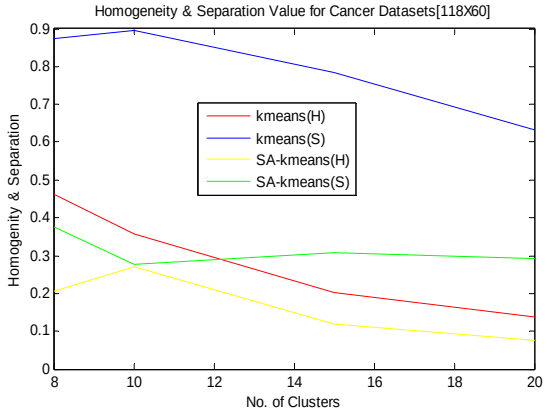

Fig. 12: Comparison of Homogeneity & Separation value for Cancer Datasets.

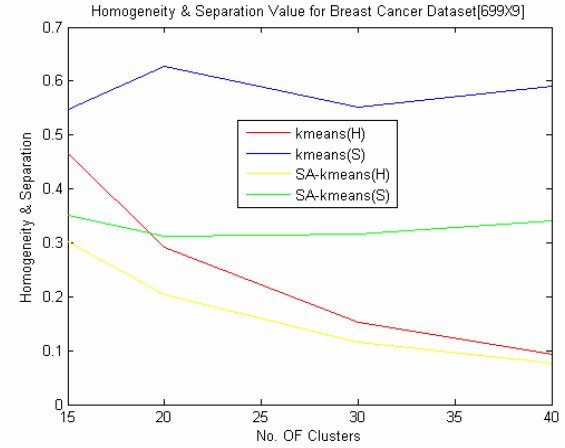

Fig. 13: Comparison of Homogeneity & Separation value for Breast Cancer Dataset

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed an efficient approach toidentifying and validating clusters in multivariate datasets. Performance experiments on real and synthetic microarray datasets showed that the proposed approach achieve a high degree of robustness, efficiency and clustering quality, compared to k-means clustering methods for gene expression mining. In the future, we will further explore the following issues:

1. We will design a memory-efficient clustering method which will be integrated into our iteratively clustering approach. This will be especially useful for very large datasets.

2. We will extend our approach to capture the pattern structure embedded in the data set. This will provide more insight into the relationships between the data points in the dataset.

VI. ACKNOWLEDGMENT

The authors would like to thank the University Medical Centre, Institute of pharmacology, Ljubljana, Yugoslavia for providing 'Breast cancer Datasets' and Dr. John N. Weinstein at laboratory of Molecular Pharmacology CCR, NCI, NIH, DHHS for providing 'Cancer Datasets'.

VII. REFERENCES

[1] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," in Proceedings of the ACM SIGMOD International Conference on Management of Data, 1998, pp. 94-105.

[2] 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," in Proceedings of National Academy of Science, Vol. 96, 1999, pp. 6745-6750.

[3] A. Ben-Dor and Z. Yakhini, "Clustering gene expression patterns," Journal of Computational Biology, Vol. 6, 1999, pp. 281-297.

[4] A. Brazma, I. Jonassen, J. Vilo, and E. Ukkonen. "Predicting gene regulatory elements in silico on a genomic scale," Genome Research, Vol. 8, 1998, pp. 1202-1215.

[5] J. DeRisi, L. Penland, P. O. Brown, M. L. Bittner, P. S. Meltzer, M. Ray, Y. Chen, Y. A. Su, and J. M. Trent, "Use of a cDNA microarray to analyze gene expression patterns in human cancer," Nature Genetics, Vol. 14, 1996, pp. 457- 460.

[6] M. B. Eissen, P. T. Spellman, P. O. Brown, and D. Botstein, "Clustering analysis and display of genome wide expression patterns," in Proceedings of the National Academy of Sciences, Vol. 95, 1998, pp. 14863-14868.

[7] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD '96), 1996, pp. 226-231.

[8] S. Guha, R. Rastogi, and K. Shim, "CURE: an efficient clustering algorithm for large databases," in Proceedings of the ACM SIGMOD International Conference on Management of Data, 1998, pp. 73-84.

[9] J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, San Francisco, 2001.

[10] A. K. Jain and R. C. Dubes, Algorithms for Clustering Data, Prentice Hall, New Jersey, 1988.

[11] O T. Kohonen, "The self-organizing map," in Proceedings of the IEEE, Vol. 78, 1990, pp. 1464-1480.

[12] Ingber, L., 1993, "Simulated annealing: practice versus theory", Mathl. Comput. Modelling 18, 11, 29-57.

[13] Deboeck, G. J. [Ed.], "Trading On The Edge", Wiley, 1994.

[14] Crama, Y., and M. Schyns, 1999, "Simulated annealing for complex portfolio selection problems."

[15] Goffe, W.L., G.D. Ferrier and J. Rogers, 1994, "Global optimisation of statistical functions with simulated annealing", J. Econometrics 60 (1/2), 65-100.

[16] Ingber, L., M.F. Wehner, G.M. Jabbour and T.M. Barnhill, 1991, "Application of statistical mechanics methodology to term-structure bond-pricing models", Mathl. Comput. Modelling 15 (11), 77-98.

VIII. BIOGRAPHY

Kumar Dhiraj, born in patna, INDIA on June 12, 1982. He received the B.E. degree in Biotechnology from the PESIT-Bangalore, VTU-Karnataka in 2006. In January 2007, he joined as Research scholar in department of computer science engineering at NIT Rourkela, INDIA. He is the author of 4 papers published in academic journals and international conference proceeding in the fields of computer science and bioinformatics. His current research interests include pattern recognition, Data mining, soft computing and bioinformatics. He received the young innovative mind-05 award organized by Honeywell Technologies.