IJESRT
INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

CLUSTERING ALGORITHMS- FOR GENE EXPRESSION ANALYSIS

Pooja Pandey*, Jai Pratap Dixit, Brijesh Kumar Pandey
* M.Tech. Student Goel Institute of Technology & Management Lucknow
AP –IT Department Ambalika Institute of Management and Technology Lucknow
AP-CS Department Goel Institute of Technology & Management Lucknow

DOI: 10.5281/zenodo.192575

ABSTRACT

Data Mining refers to as the nontrivial process of “identifying valid, novel, potentially useful and ultimately understandable pattern in data". Based on the type of knowledge that is mined, data mining can be classified in to different models such as Clustering, Decision trees, Association rules, and Sequential pattern and time series. In this paper work, an attempt has been made to study theoretical background and applications of Clustering techniques in data mining with a special emphasis on analysis of Gene Expression under Bioinformatics.

Bioinformatics is the study of genetic and other biological information using computer and statistical techniques. DNA microarray technology has now made it possible to simultaneously monitor the expression levels of thousands of genes during important biological processes and across collections of related samples.

1) A good of data means that many of the challenges in biology are now challenges in computing.
2) A step toward addressing this challenge is the use of clustering technique, which is essential in the data mining process to reveal natural structures and identifying interesting patterns in the underlying data.

In this paper work, effort has been made to compare between few Clustering algorithms such as: K means, Hierarchical, Self-Organization Map (SOM), and Cluster Affinity Search Technique (CAST) with proposed algorithm called CAST+. Strengths and Weaknesses of the above Clustering algorithms are indented and drawbacks like knowing number of clusters before clustering, and taking affinity threshold as input from the users are rectified by the proposed algorithm. Results show that Proposed Algorithm is efficient in comparison with other Clustering algorithms mentioned above.

INTRODUCTION

With the enormous amount of data stored in Files, databases, and other repositories, it is increasingly important, to develop powerful means for analysis and perhaps interpretation of such data and for the extraction of interesting knowledge that could help in decision- making.

Data Mining, also popularly known as Knowledge Discovery in Databases (KDD), refers to as “The nontrivial process of identifying valid, novel, potentially useful and ultimately understandable pattern in data". While data mining and knowledge discovery in databases (KDD) are frequently treated as synonyms, data mining is actually part of the knowledge discovery process. Figure 1.1 shows data mining as a step in an iterative knowledge discovey process.

The task of the knowledge discovery and data mining process is to extract knowledge from data such that the resulting knowledge is useful in a given application. The Knowledge Discovery process in Databases comprises of a few steps leading from raw data collections to some form of retrieving new knowledge. The iterative process consists of the following steps:

***Data cleaning*:** Also known as data cleansing, it is a phase in which noisy data and irrelevant data are removed from the collection.

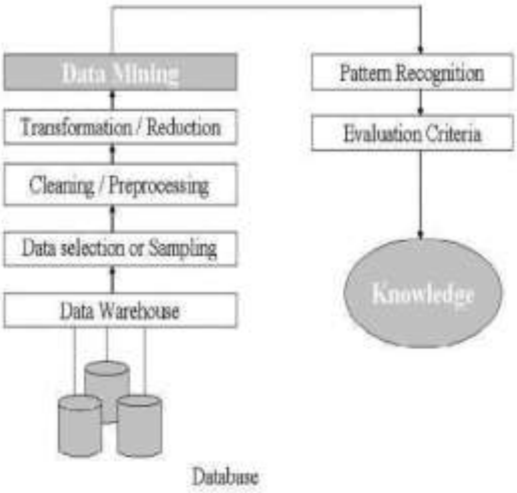


Figure 1.1: An Overview of the Steps Comprising the KDD Process

Data integration: At this stage, multiple data sources, often heterogeneous, may be combined in a common source.
Data selection: At this step, the data relevant to the analysis is decided on and retrieved from the data collection.
Data mining*:* It is the crucial step in which clever techniques are applied to extract data patterns potentially useful.
Pattern evaluation: In this step, strictly interesting patterns representing Knowledge is identified based on given measures.
Knowledge representation: Is the final phase in which the discovered knowledge is visually represented to the user.
This essential step uses visualization techniques to help users understand and interpret the data mining results.

MICROARRY TECHNOLOGY

Compared with the traditional approach to genomic research, which is focused on the local examination and collection of data on single genes, microarray technologies have now made it possible to monitor the expression levels for tens of thousands of genes in parallel. The two major types of microarray experiments are the cDNA microarray and oligonucleotide arrays (abbreviated olio chip). Despite differences in the details of their experiment protocols, both types of experiments involve three common basic procedures:

1) *Chip manufactu*re**:** A microarray is a small chip (made of chemically coated glass, nylon membrane or silicon), onto which tens of thousands of DNA molecules (probes) are attached indexed grids. Each grid cell relates to a DNA sequence.
2) *Target preparation*, labeling and hybridization: Typically, two mRNA samples (a test sample and a control sample) are reverse-transcribed into cDNA (targets), labeled using either fluorescent dyes or radioactive isotopic, and then hybridized with the probes on the surface of the chip.
3) *The scanning process:* Chips are scanned to read the signal intensity that is emitted from the labeled and hybridized targets.
4) Generally, both cDNA microarray and olio chip experiments measure the expression level for each DNA sequence by the ratio of signal intensity between the test sample and the control sample, therefore, data sets resulting from both methods share the same biological semantics. In this paper work, unless explicitly stated, we will refer to both the cDNA microarray and the oligo chip as microarray technology and term the measurements collected via both methods as gene expression data.

Gene expression data: A microarray experiment typically assesses a large number of DNA sequences (genes, cDNA clones, or expressed sequence tags) under multiple conditions. These conditions may be a time series during a biological process or a collection of different tissue samples (e.g., normal versus cancerous tissues). In this paper work, emphasis is given on the cluster analysis of gene expression data without making a distinction among DNA sequences, which will uniformly be called "genes". Similarly, it is referred to all kinds of experimental conditions as "samples", if no confusion will be caused. A gene expression data set from a microarray experiment can be represented by a real-valued expression matrix *M = fWij j*

1 · *i* · *n;* 1 · *j* · *mg* as shown in Figure 1.2,

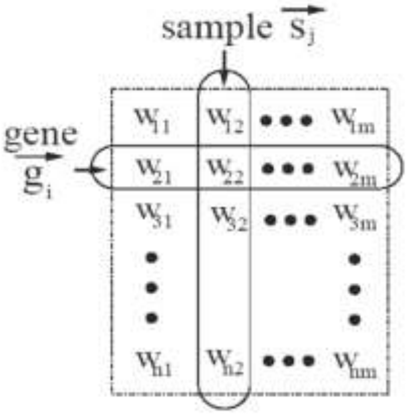

sample $\vec{s_j}$
gene
$\vec{g_i}$
w_{11} w_{12} ••• w_{1m}
w_{21} w_{22} ••• w_{2m}
w_{31} w_{32} ••• w_{3m}
w_{n1} w_{n2} ••• w_{nm}

Fig. 1.2: A Gene Expression Matrix

where the rows (*G* = *fg*1*:::gng*) form the expression patterns of genes, the columns (*S* = *fS*1*:::Smg*) represent the expression pro¯les of samples, and each cell is the measured expression level of gene i in sample j. Table 1.1 includes some notation that is used in paper work.

n	**number of genes**
m	**number of samples**
M	**a gene expression matrix**
wij	**each cell in gene expression matrix**
gi	**a gene**
Sj	**a sample**
G; G0; :::	**a set of genes**
S; S0; ::::	**a set of samples**

Table 1.1: Notation in this Paper

LITERATURE REVIEW

According to *Reichhardt T(1999)* , Biological data are being produced at a phenomenal rate [31]. For example as of April 2001, the GenBank repository of nucleic acid sequences contained 1,15,46,000 entries and the SWISSPROT database of protein sequences contained 95,320 entries. On an average, these databases are doubling in size every 15 months. In addition, since the publication of the H. influenza genome [14], complete sequences for nearly 300 organisms have been released, ranging from 450 genes to over 100,000. At the same time, there have been major advances in the technologies that supply the initial data.

Anthony Kervalage of Celera recently cited that an experimental laboratory can produce over 100 gigabytes of data a day with ease [20]. Figure 2.1 shows the growth of DNA sequence in Gen Bank during a period from 1982 to 2003. This incredible processing power has been matched by developments in computer technology; the most important areas of improvements have been in the CPU speed, disk storage and Internet, allowing faster computations, better data storage and revolutionaries the methods for accessing and exchanging data.

According to *Dr. Diego Kuonen*, Data mining and bioinformatics are fast expanding research frontiers. It is important to examine what are the important research issues in bioinformatics and develop new data mining methods for scalable and effective analysis. The effective interactions and collaborations between these two Felds have just started and lots of exciting results will appear in the feature. Bioinformatics and Data mining will inevitably grow toward each other because bioinformatics will not become knowledge discovery without statistical datamining and thinking[8].

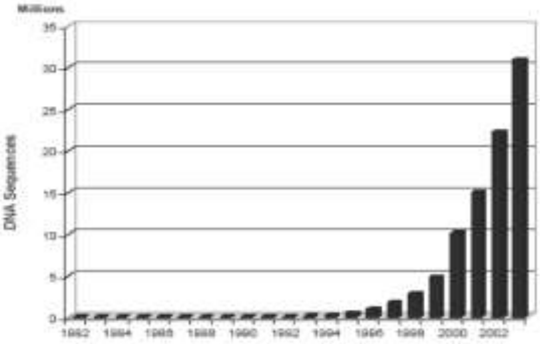


Figure 2.1 : Figure Showing the growth of gen bank .

According to *P.Tamayo*, The main types of data analysis needed to for biomedical applications include:

- *Clustering:* Finding new biological classes or refining existing ones [10].
- *Gene Selection:* In mining terms this is a process of attribute selection, which Finds the genes most strongly related to a particular class.
- *Classification:* classifying diseases or predicting outcomes based on gene expression patterns, and perhaps even identifying the best treatment for given genetic signature.

One important clustering task is to identify groups of co expressed genes recognize coherent expression patterns. Microarrays are a revolutionary new technology with great potential to provide accurate medical diagnostics, help Find the right treatment and cure for many diseases and provide a detailed genome-wide molecular portrait of cellular states . *Houle et al. (2000)* refer to a classification of three successive levels for the analysis of biological data, that is identified on the basis of the central dogma of molecular biology: Application of Data Mining techniques for Bioinformatics is vast area to study. It includes [11]

- *Gene expression in Datamining:* Gene expression analysis is the use of quantitative mRNA-level measurements of gene expression (the process by which a gene's coded information is converted into the structural and functional units of a cell) in order to characterize biological processes and elucidate the mechanisms of gene transcription.
- *Data mining in genomics***:** Genomics is the study of an organism's genome and deals with the systematic use of genome information to provide new biological knowledge.
- *Data mining in proteomics:* Proteomics is the large-scale study of proteins, particularly their structures and functions.

Clustering techniques have proven to be helpful to understand gene function, gene regulation, cellular processes, and subtypes of cells[3]. Genes with similar expression patterns(co-expressed genes) can be clustered together with similar cellular functions. This approach may further understanding of the functions of many genes for which information has not been previously available [13]. Furthermore, co-expressed genes in the same cluster are likely to be involved in the same cellular processes, and a strong correlation of expression patterns between those genes indicates co-regulation. Searching for common DNA sequences at the promoter regions of genes within the same cluster allows regulatory motifs specific to each gene cluster to be identified and regulatory elements to be proposed [15]. The inference of regulation through the clustering of gene expression data also gives rise to hypotheses regarding the mechanism of the transcriptional regulatory network. Finally, clustering different samples on the basis of corresponding expression profiles may reveal sub-cell types which are hard to identify by traditional morphology-based approaches.

According to *Kjersti Aas*[18], DNA microarray makes it possible to quickly, efficiently and accurately measure the relative representation of each mRNA species in the total cellular mRNA population. A DNA experiment consists of measurements of the relative representation of a large number of mRNA species (typically thousands or tens of thousands) in a set of related biological samples, e.g. time points taken during a biological process or clinical samples taken from different patients. Each experiment sample is compared to a common reference sample and the result for each gene is the ratio of the results of such experiments are represented in a table, with each row representing a gene, each column a sample, and each cell the log(base - 2) transformed expression ratio of the appropriate gene in the appropriate sample.

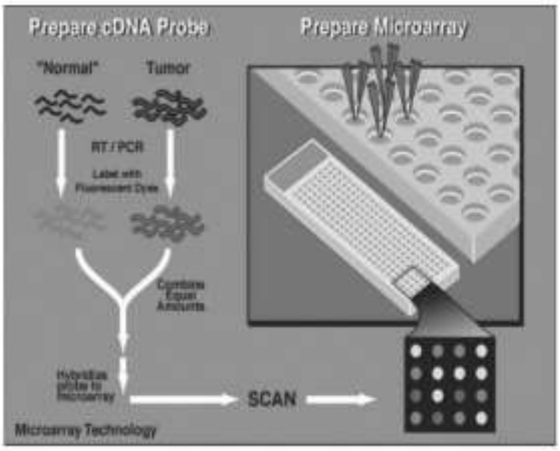


Figure 2.2: Explaining the micro array process

The microarray process is shown in Figure 2.2. The DNA sample (which may be several thousands) are Fixed to a glass slide, each at a known position in the array. A target sample and a reference sample are labeled with red and green dyes, respectively, and each is hybridized on the slide. Using a Fluorescent microscope and image analysis, the log(green/red) intensities of mRNA hybridizing at each site is measured. The result is a few thousand numbers, typically ranging form -4 to 4, measuring the expression level of each gene in the experimental sample relative to the reference sample. Positive values indicate higher expression in the target versus the reference, and vice verses for negative values.

According to *Rui Xu, and Donald Wunsch* [27], data analysis plays an indispensable role for understanding various phenomena. Cluster analysis, primitive exploration with little or no prior knowledge, consists of research developed across a wide variety of communities. Cluster analysis is not a one-shot process. It needs a series of trials and repetitions. Moreover, there are no universal and effective criteria to guide the selection of features and clustering schemes. Validation criteria provide some insights on the quality of clustering solutions. But how to choose the appropriate criterion is still a problem, which requires more efforts. Clustering has been applied in a wide variety of Fields, ranging from engineering (machine learning, artificial intelligence, pattern recognition, mechanical engineering, electrical engineering), computer sciences (web mining, spatial database analysis, textual document collection, image segmentation), life and medical sciences (genetics, biology, micro biology, paleontology, psychiatry, clinic, pathology), to earth sciences (geography. geology, remote sensing), social sciences (sociology, psychology, archeology), and economics (marketing, business).

Accordingly, clustering is also known as numerical taxonomy, learning without a teacher (or unsupervised learning), typological analysis and partition. The diversity reflects the important position of clustering in scientific research. On the other hand, it causes con- fusion, due to the differing terminologies and goals. Clustering algorithms developed to solve a particular problem, in a specialized Field, usually make assumptions in favor of the application of interest. These biases inevitably affect performance in other problems that do not satisfy these premises.

CLUSTER FORMATION ALGORITHMS

Clustering plays a vital role in the Gene Expression Analysis. In this chapter we will first discuss the concepts of *clustering*, and later discuss the various algorithms used such as K-Means, SOM, hierarchical clustering algorithms and their pros and cons.

Cluster formation in Data Mining: Clustering is the process of grouping data objects into a set of disjoint classes, called *clusters*, so that objects within the same class have high similarity to each other, while objects in separate classes are more dissimilar. Clustering is an example of *unsupervised classification*. “Classification" refers to a procedure that assigns data objects to a set of classes. “Unsupervised" means that clustering does not rely on predefined classes and training examples while classifying the data objects. Thus, clustering is distinguished from pattern recognition or the areas of statistics known as discriminate analysis and decision analysis, which seek to Find rules for classifying objects from a given set of pre-classified objects.

Categories of Gene Expression Data Clustering: Recently, a typical micro array experiment contains 103 to 104 genes, and this number is expected to reach the order of 106. However, the number of samples involved in micro array experiment is generally less than 100. One of the characteristics of gene expression data is that it is meaningful to cluster both genes and samples. Clustering gene expression data can be categorized into two groups.
Gene based clustering : In this type of clustering genes are treated as the objects, while samples as the features. The purpose of gene-based clustering is to group together co expressed genes which indicate co-function and co-regulation.

Sample based clustering: In this type of clustering samples are the objects and genes are features. Within a gene expression matrix, there are usually particular macroscopic phenotypes of samples related to some diseases or drug effects, such as diseased samples, normal samples or drug treated samples. The goal of sample based clustering is to Find the phenotype structures or sub- structures of the sample.
Proximity measurement for gene expression data : Proximity measurement measures the similarity(distance) between two data objects. Gene expression data objects, no matter genes or samples, can be formalized as numerical vectors

Oi = *foi;jj*1 · *j* · *pg;* where *oi;j*

However, for gene expression data, the overall shapes of gene expression patterns are of greater interest than the individual magnitudes of each feature.
object and *p* is the number of features. The proximity between two objects *Oi* and *Oj* is
measured by a *proximity function* of corresponding vectors *Oi* and *Oj*.
Euclidean Distance: Euclidean Distance is one of the most commonly used methods to measure the distance

between two data objects. The distance between
objects *Oi* and *Oj* in p-dimensional space
is defined as

Eucledian(*Oi;* *Oj*)
= v
p (*oid* *ojd*)2 (3.1)
u *¡*
uX
t
d=1

Pearson's correlation coefficient : Pearson's correlation coeffcient, which measures the similarity between the shapes of two expression patterns. Given two data objects *Oi* and *Oj*, pearson's correlation coe±cient is defined as

pearson(*Oi; Oj*) = *dp*=1(*oid ¡* ^{1}*oi*)(*ojd ¡* ^{1}*oj*) (3.2)
p (*oid* ^{1}*oi*)2 *p* (*ojd* ^{1}*oj*)2

=1
d P *d*=1
pP *¡* pP *¡*
~ ~

where ^{1}*oi* and ^{1}*oj* are the means for *Oi* and *Oj* respectively. Pearsons correlation coefficient
cient views each object as a random variable with p observations and measures the similarity between two objects by calculating the linear relationship between the distributions of the two corresponding random variables[16].
Hence, in this entire paper work Euclidean distance is used as the proximity measure.
Clustering Paradigms : Based on the method of clustering, the clustering algorithms are divided into two paradigms:
Partitioning clustering: in which the database is partitioned into a predefined number of clusters.

for example: K-Means, K-mediods etc. Hierarchical clustering do sequence of partitions, in which each partition is nested into the next partition in the sequence Based on the approach Hierarchical clustering is further divided into two types

***Agglomerative clustering**:* technique starts with as many clusters as there are records, with each cluster having only one record. Then pair of clusters successively merged. This is also called as bottom up approach.

for example: single linkage hierarchical algorithm, complete linkage hierarchical algorithm etc.

***Divisive clustering**:* takes the opposite approach from agglomerative techniques. In this approach the algorithm starts with one cluster containing all the data objects, and at each step split a cluster until only singleton clusters of individual objects remain.

for example: Graph theoretical algorithms, CAST etc.

Clustering Algorithms : In this section different algorithms that is studied in this paper work is discussed in brief.

K-Means Algorithms : The K-Means algorithm is a typical partition-based clustering method. Given a pre-specified number K, the algorithm partitions the data set into K disjoint subsets which optimize the following objective function:

k

$E =$ jO ¡ $^{1}ij2:$ (3.3)
$i=1$ $O^{2}C$

Here, O is a data object in cluster *Ci* and *mui* is the centroid (mean of objects) of *Ci*. Thus, the objective function E tries to minimize the sum of the squared distances of objects from their cluster centers.

Algorithm:

1. The K-Means algorithm accepts the "number of clusters" to group data into, and the dataset to cluster the input values.
2. The K-Means algorithm then creates the first k initial clusters from the data set
3. The K-Means algorithm calculates the arithmetic mean of each cluster formed in the data set. The arithmetic mean is the mean of all the individual records in the cluster.
4. Next K-Means assigns each record in the dataset to only one of the initial clusters. Each record is assigned to the nearest cluster using proximity measure like Euclidean distance.
5. K-Means reassigns each record in the dataset to the most similar cluster and recalculates the arithmetic mean of the clusters in the dataset.
6. K-Means reassigns each record in the dataset to only one of the new clusters formed
7. The preceding steps are repeated until "stable clusters" are formed and the K-Means clustering is completed

The K-Means algorithm is simple and fast. The time complexity of K-Means is O(l*m*n), where *l* is the number of iterations and K is the number of clusters, *m* is the number of genes and *n* is the number of samples. Our empirical study has shown that the K-Means algorithm typically converges in a small number of iterations. However, it also has several drawbacks as a gene-based clustering algorithm. First, the number of gene clusters in a gene expression data set is usually unknown in advance. To detect the optimal number of clusters, users usually run the algorithms repeatedly with different values of K and compare the clustering results.

For a large gene expression data set which contains thousands of genes, this extensive parameter ¯ne-tuning process may not be practical. Second, gene expression data typically contain a huge amount of noise; however, the K-Means algorithm forces each gene into a cluster, which may cause the algorithm to be sensitive to noise.

SOM

The Self-Organizing Map (SOM) was developed by Kuonen in 1997[12], on the basis of a single layered neural network. The data objects are presented at the input, and the output neurons are organized with a simple neighborhood structure such as a two dimensional p*q grid. Each neuron of the neural network is associated with a reference vector, and each data point is “mapped" to the neuron with the “closest" reference vector.

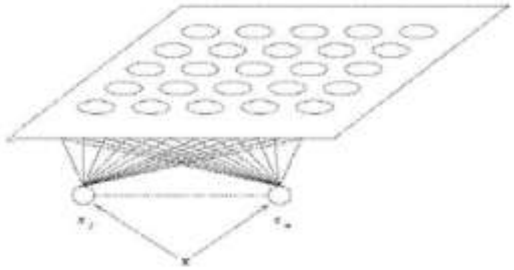


Fig 3.1 : Schematic Representation of a self-organizing map method

The neuron training process of SOM provides a relatively more robust approach than K-Means to the clustering of highly noisy data [12]. However, SOM requires users to input the grid structure of the neuron map. This parameter is preserved through the training process; hence, improperly-specified parameter will prevent the recovering of the natural cluster structure. Furthermore, if the data set is abundant with irrelevant data points, such as genes with invariant patterns, SOM will produce an output in which this type of data will populate the vast majority of clusters. In this case, SOM is not effective because most of the interesting patterns may be merged into only one or two clusters and cannot be identified.

Hierarchical Clustering : *Hierarchical clustering* generates a hierarchical series of nested clusters which can be graphically represented by a tree, called *dendrogram*. The branches of a dendrogram not only record the formation of the clusters but also indicate the similarity between the clusters. By cutting the dendrogram at some level, we can obtain a specified number of clusters. By reordering the objects such that the branches of the corresponding dendrogram do not cross, the data set can be arranged with similar objects placed together.
The hierarchical clustering scheme:
Let S=*fSi ; jg* is the input similarity matrix, where *Si ; j* indicates similarity between two data objects based on Euclidean distance.
Algorithm:
1.Find a minimal entry s(i, j) in S, and merge clusters i and j.
2.Modify S by deleting rows and columns i, j and adding a new row *i* and column *j*, with their dissimilarities defined by:

$s(k; i [j) = s(i [j; k) = is(k; i) + js(k; j) + js(k; i) ¡ s(k; j)j$ (3.4)
3. If there is more than one cluster, then go to Step 1.
Common variants of this scheme, obtained for appropriate choices of the ¡ *s* and °parameters, are the following:

Single linkage : $s(k; i [j) = min = fs(k; i); s(k; j)g$ (3.5)

Complete linkage : $s(k; i [j) = maxfs(k; i); s(k; j)g$ (3.6)

Average linkage : $s(k; i [j) = (nid(k; i) + njd(k; j))=(ni + nj);$ (3.7)
where *ni* denotes the number of elements in cluster i.

Hierarchical clustering not only groups together genes with similar expression pattern but also provides a natural way to graphically represent the data set. The graphic representation allows users a thorough inspection of the whole data set and obtain an initial impression of the distribution of data. However, the conventional agglomerative approach suffers from a lack of robustness [19] , i.e., a small perturbation of the data set may greatly change the structure of the hierarchical dendrogram. Another drawback of the hierarchical approach is its high computational complexity. To construct a complete dendrogam (where each leaf node corresponds to one data object, and the root node corresponds to the whole
data set), the clustering process should take *n*2¡*n* merging (or splitting) steps. The time
n
complexity for a typical agglomerative hierarchical algorithm is *O*(*n*2*logn*) [17]. If a wrong decision is made in the initial steps, it can never be corrected in the subsequent steps.

PROPOSED ALGORITHM

Several data mining solutions have been presented for Bioinformatics [22], [23], and [5]. Cluster analysis received significant attention in the area of gene expression. It allows the identification of groups of similar objects in multidimensional space. In this chapter it has been discussed a graph-based clustering algorithm, CAST, its disadvantages, and how they are overcome in the proposed algorithm.

Cluster Affinity Search Technique(CAST): On study of clustering algorithms with an emphasis on graph theoretic approaches[21], it is observed that for any micro array data analysis of gene expression patterns with clustering algorithms involve the following steps:

***Determination of gene expression data*:** The data can be represented by a real- valued *expression matrix I* where *Iij* is the measured expression level of gene *i* in experiment *j*. The *ith* row of the matrix is a vector forming the *expression pattern* of gene *i*.

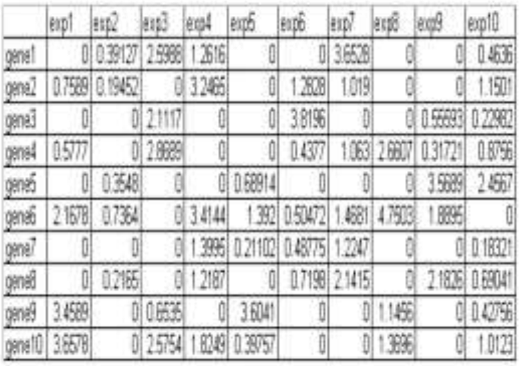
Calculation of a similarity matrix S: In this matrix, the entry *Sij* represents the similarity of the expression patterns for genes *i* and *j*. Many possible similarity measures can be used here. A good choice of measure depends on the nature of the biological question and on the technology that was used to obtain the data.

Clustering the genes based on the similarity data or the expression data: Genes that belongs to the same cluster should have similar expression patterns, while different clusters should have distinct, well-separated patterns.

Representation of the constructed solution : Several techniques were previously used in clustering gene expression data such as Hierarchical clustering techniques[13], Self-Organizing Maps used by Tamayo et. al [12], and K-Means [1]. In this paper work it has been discussed a novel algorithm for the problem of clustering gene expression patterns. Unlike the hierarchical approaches mentioned above, our algorithm doesn't build a tree of clusters. Clusters are built and portrayed as unrelated entities. In contrast to self-organizing maps, it does not assume an initial spatial structure, but determines the cluster and structure based on the data. Unlike K-Means it doesn't require the number of clusters earlier before clustering.

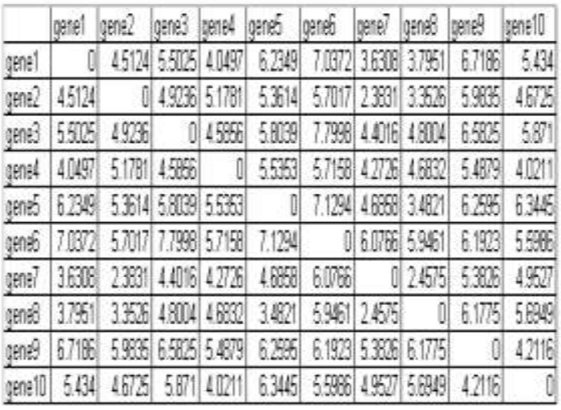
Experimental Representation of Data set : Formally, a set of genes can be viewed as a set of vectors *V* = *fvi; v*2*; v*3*; :::; vmg* with each expression level of a given experiment, *xj*, being the components in the vector *vi*= (*x*1*; x*2*; x*3*; ; xn*), where *m* is the number of genes in the experiments and *n* is the number of experiments. Table 4.1 is an example gene expression matrix.(This works equally well when the experiments form the vectors).

These vectors can then be viewed as points in n dimensional space and a similarity measurement between points can be calculated and stored in a *m by m* similarity matrix *M*. Where *Mij* is the distance (similarity) measure between gene *i* and gene *j*. There are several similarity measures, e.g., Euclidean distance and Pearson correlation.



	exp1	exp2	exp3	exp4	exp5	exp6	exp7	exp8	exp9	exp10
gene1	0	0.39127	2.5988	1.2616	0	0	3.6528	0	0	0.4636
gene2	0.7589	0.19452	0	3.2465	0	1.2828	1.019	0	0	1.1501
gene3	0	0	2.1117	0	0	3.8196	0	0	0.55693	0.22982
gene4	0.5777	0	2.8689	0	0	0.4377	1.063	2.6607	0.31721	0.8756
gene5	0	0.3548	0	0	0.68914	0	0	0	3.5689	2.4567
gene6	2.1678	0.7364	0	3.4144	1.392	0.50472	1.4681	4.7503	1.8895	0
gene7	0	0	0	1.3995	0.21102	0.48775	1.2247	0	0	0.18321
gene8	0	0.2165	0	1.2187	0	0.7198	2.1415	0	2.1826	0.69041
gene9	3.4589	0	0.6535	0	3.6041	0	0	1.1456	0	0.42756
gene10	3.6578	0	2.5754	1.8249	0.39757	0	0	1.3896	0	1.0123

Table 2.1: A Gene Expression Matrix



	gene1	gene2	gene3	gene4	gene5	gene6	gene7	gene8	gene9	gene10
gene1	0	4.5124	5.5025	4.0497	6.2349	7.0372	3.6308	3.7951	6.7186	5.434
gene2	4.5124	0	4.9236	5.1781	5.3614	5.7017	2.3831	3.3526	5.9835	4.6725
gene3	5.5025	4.9236	0	4.5866	5.8039	7.7998	4.4016	4.8004	6.5825	5.871
gene4	4.0497	5.1781	4.5866	0	5.5353	5.7158	4.2726	4.6832	5.4879	4.0211
gene5	6.2349	5.3614	5.8039	5.5353	0	7.1294	4.6858	3.4821	6.2595	6.3445
gene6	7.0372	5.7017	7.7998	5.7158	7.1294	0	6.0766	5.9461	6.1923	5.5986
gene7	3.6308	2.3831	4.4016	4.2726	4.6858	6.0766	0	2.4575	5.3826	4.9527
gene8	3.7951	3.3526	4.8004	4.6832	3.4821	5.9461	2.4575	0	6.1775	5.6949
gene9	6.7186	5.9835	6.5825	5.4879	6.2595	6.1923	5.3826	6.1775	0	4.2116
gene10	5.434	4.6725	5.871	4.0211	6.3445	5.5986	4.9527	5.6949	4.2116	0

Table 2.2: Similarity Matrix For the above gene expression matrix

CAST : The Cluster affnity search technique (CAST) developed by Ben-Dor et. al., 1999 [2] takes a graph theoretic approach that relies on the concept of a clique graph and uses a divisive clustering approach. A clique graph is an undirected graph that is the union of disjoint complete graphs. Thus, the model assumes that there is a true biological partition of the genes into disjoint clusters bases on the functionality of the genes. The clique graph would then be composed of clusters (cliques) of genes (vertices) whose interconnections (edges) are present or not present corresponding to their respective similarity measures (i.e. if two genes are similar there is an edge between them). So, ideally, the genes would form sub graphs (cliques) where every gene would be completely similar to every other gene in the clique and completely dissimilar to every gene not in the clique.

It is very probable that a set of gene (or experiment) vectors will tend to have a similarity gradient across other vectors and the high incidence rate of errors inmicro-array technology, the ideal clique graph would be impossible to generate, or, at the very lease, would create very small clusters. So small, in fact, that many would contain single data points, and therefore defeat the purpose of the algorithm. Thus, an approximation of the preceding model is called for.

It is studied that the main draw back of CAST algorithm is taking *a±nity threshold* as input, which determines the size and number of clusters produced. In this paper work we have proposed an *affnity threshold* by taking the mean of affnity values of all the elements in the dataset.The proposed algorithm may be named as CAST+. Let us see the main terminology used

Definition 1: The affnity of a node x to a cluster C is defined as follows:

$$a(x) = \quad S(x; k) \qquad (4.1)$$

$k^{2}C$

26

Definition 2:The connectivity threshold,$\hat{A}$, of a cluster C is:

$$\hat{A} = T\,j\,C\,j \qquad (4.2)$$

where $j\,C\,j$ is the cardinality of C.

Definition 3: A high connectivity node is a node that will be included in a cluster. Its affinity satisfies the following.

$$a(i) \text{ ¸ } \hat{A} \qquad (4.3)$$

where a(i) is the affinity of i.

Definition 4: A low affinity node will be removed from a cluster if its affinity satisfies the following:

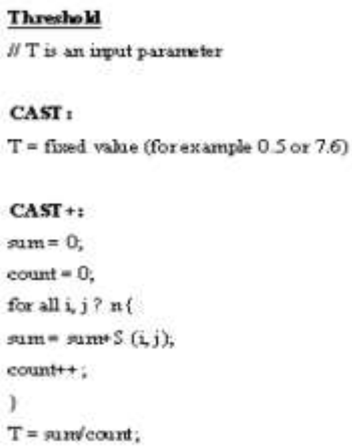
$$a(i) < \hat{A} \qquad (4.4)$$

where a(i) is the affinity of i.

Each cluster is formed by alternating between adding and removing nodes from the current cluster until such time that changes no longer occur or a maximum of iterations executed:

Node Addition: Add nodes with high connectivity to the nodes in the open cluster.

For CAST+ Before performing this step we check the node with existing clusters and adds to the cluster which is highly connected.
Node Removal: Remove any nodes in the open cluster with low connectivity to the other nodes in the cluster.
Cluster Cleaning: Make sure all nodes are in clusters with highest affinity.
For CAST+ this step is not required.



```
Threshold
// T is an input parameter

CAST :
T = fixed value (for example 0.5 or 7.6)

CAST+:
sum = 0;
count = 0;
for all i, j ? n {
sum= sum+S (i, j);
count++ ;
}
T = sum/count;
```

.

Pseudo code for finding threshold Value

The threshold assignment and affinity check with existing clusters in the step of node addition, obviate the need for the “cleaning " step as proposed in the original CAST algorithm. The cleaning step is used to move any vector from its current cluster to one that it may have a higher afnity for and has a time complexity on the order of *O*(*n*2). The output of the gene expression matrix of Figure 4.1 is given in Figure 4.7

Analysis of Clustering Solutions : Different clustering algorithms yield different solutions on the same data and also same algorithm gives different solutions for different parameter settings.

Different measures for the quality of a clustering solution are applicable in different situations, depending on the data and on availability of the true solution.
In case true solution is known, and we wish to compare it to another solution, one can use Murkowski Measure or Jacquards Coefficient method.

Jaccards Coeffcient : Jaccards Coeffcient is a static measure used for comparing the similarity and diversity of sample sets, by everitt(1993)[13].
The jaccards coefficient is defned as the size of the intersection divided by the size of the union of the sample sets.

$$J(A; B) = \frac{j\,A \backslash B\,j}{j\,A\,[\,B\,j} \qquad (4.6)$$

where A indicates the true solution, and B indicates the solution generated by the algorithm.

Minowski Measure : A clustering solution of *n* elements is represented by a *n ¡ by ¡ n* similarity matrix C, where *Cij* = 1 if i and j belong to the same cluster and *Cij* = 0 otherwise.

Given such matrix representation of the true clustering *T* and any clustering *C* of the same data set, the minowski measure for the quality of *C* is the normalized distance between
the two matrices

$$MinowskiMeasure = \frac{j\,T\,¡\,C\,j}{j\,T\,j} \qquad (4.7)$$

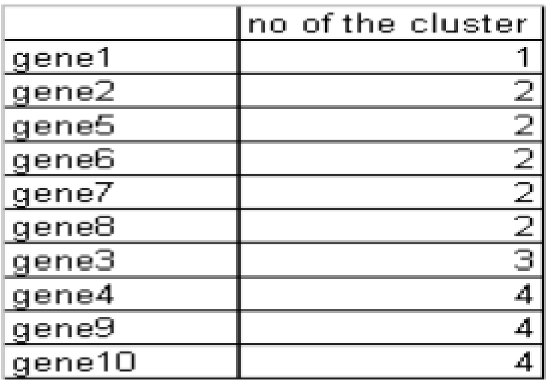
where *j T j*= q *i j* ~~*T*~~*ij*2 as developed by Sokal(1977)

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

~~P P~~
[33].

Since the matrices are binary, this is simply the number of pairs on which the two solutions disagree and normalized according to the true solution. A perfect clustering would thus obtain the score zero.

Silhouette Width : The Silhouette validation technique (Rousseau w, 1987) [33] calculates the silhouette width for each sample, average Silhouette width for each cluster and overall average silhouette



	no of the cluster
gene1	1
gene2	2
gene5	2
gene6	2
gene7	2
gene8	2
gene3	3
gene4	4
gene9	4
gene10	4

Table 4.7: Sample result of the proposed algorithm

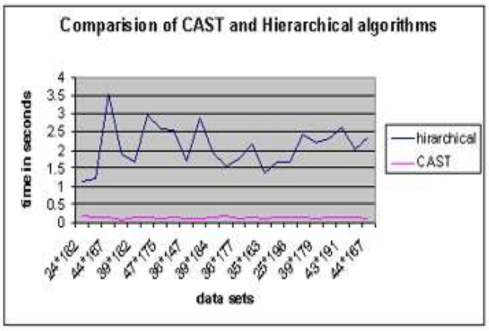
width for a total data set. Using this approach each cluster could be represented by so called silhouette width, which is based on the comparison of its tightness and separation.

The average silhouette width could be applied for evaluation of clustering validity and can also be decide how good is the number of selected clusters.
To construct the silhouettes S(i) the following formula is used:

$$S(i) = \frac{(b(i)\ ¡\ a(i))}{maxfa(i);\ b(i)g} \qquad (4.8)$$

where
a(i)=average dissimilarity of object *i* to all other objects in the same cluster.
b(i)= minimum of average dissimilarity of object *i* in other cluster (in the closest cluster).

The largest overall average silhouette indicates the best clustering. Therefore, the number of clusters with maximum overall average silhouette width is taken as the optimal number of the clusters.



Results

Fig 4.8 :Figure Showing The Comparison between the CAST and Hierarchical Algorithm

Data Set: The synthetic random datasets in our simulation provides randomly generated classes in a two dimensional Euclidean space.

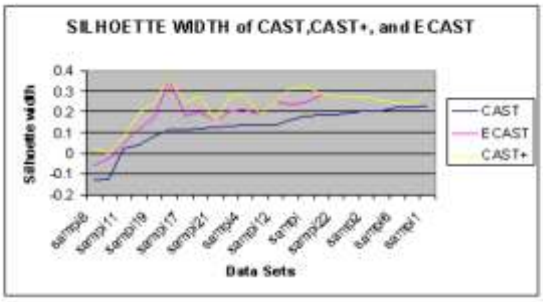


Fig 4.9 : Figure Shows the comparision between CAST , ECAST and Proposed Algorithm .

Figure 4.10 explains that CAST, ECAST, and proposed CAST+ algorithms are tested on twenty two different data sets and result is analyzed using silhouette width. It is observed that overall 22 data sets both ECAST and CAST+ algorithms performed better than CAST. But CAST+ has performed better on fourteen dataset in comparison with ECAST algorithm. In remaining data sets both have performed comparatively well. i.e. CAST+ showed 60% better performance over ECAST and 100.

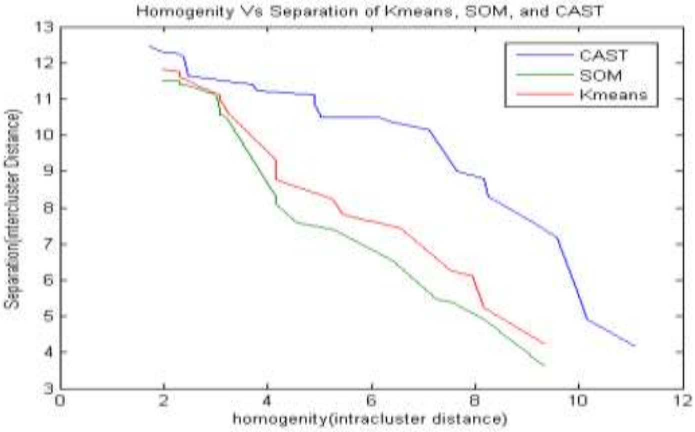


Fig .411 : Figure Showing The comparison between the K- Means ,SOM and CAST

Figure 4.11 explains the that K-Means, SOM, and CAST algorithms are performed on twenty two different data sets and respective homogeneity and separation values are calculated. It is observed that CAST has shown the best result over SOM andK-Means as indicated in Figure 4.11.

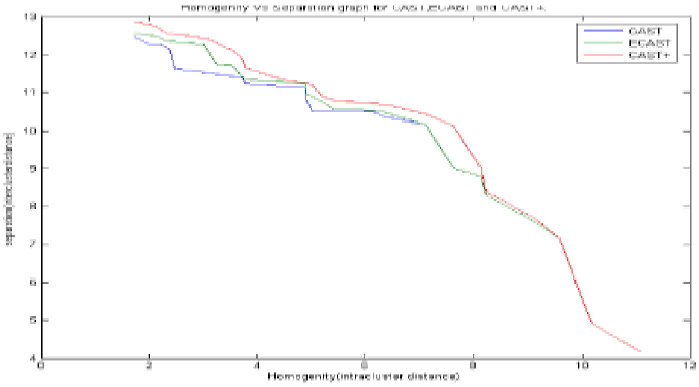


Fig 4.12 : Figure Showing the Comparison between the CAST, ECAST and proposed Algorithm .

Figure 4.12 shows that over 78% of the data sets show better result to CAST+ over CAST and ECAST. The data sets whose homogeneity value is vast, they show same result of the CAST. Overall it is observed that the proposed CAST+ algorithm shows better result than other algorithms.

CONCLUSION

In this chapter a graph theoretic divisive algorithm called CAST is studied and overcome the drawback what it is having by using the proposed algorithm. Comparing the result of the proposed algorithm with the existing algorithms and it is observed that proposed algorithm performed better than all other algorithms.

Recently, the collection of biological data has been increasing at explosive rates due to improvements of existing technologies and the introduction of new ones such as the microarrays. These technological advances have assisted the conduct of large scale experiments and research programs. The explosive growth in the amount of biological data demands the use of computers for the organization for its maintenance and the analysis. This led to the evolution of bioinformatics, an interdisciplinary field at the intersection of biology, computer science, and information technology.

Cluster analysis seeks to partition a given data set into groups based on specific features so that points within a group are more similar to each other than the points in different groups. Many conventional clustering algorithms have been adapted or directly applied to gene expression data. But each method has short comings. These shortcomings include problems of cluster boundaries, as for hierarchical techniques, where the output is a tree depicting the relation of each object to every other object in the data set. The requirement for knowing the expected number of clusters, as for K-Means, and knowing the grid structure for SOM are the underlining problems under different algorithms developed so far.

FUTURE WORK

This work can be extended as follows:

1. Improve the performance of the algorithm using soft computing techniques.
2. Perform theoretical analysis of the determination of the threshold parameter.
3. Explore further improvements to Proposed CAST+ Algorithm.

REFERENCES

[1] Arun K Pujari, Data mining Techniques, University Press, Hyderabad, 2002.

[2] Ben-Dor A., Friedman N. and Yakhini Z, "Clustering gene expression patterns",*Jour- nal of Computational Biology*, Vol. 6, No. (3/4), 1999, pp:281297.

[3] Daxin Jiang, Chun Tang and Aidong Zhang, "Cluster Analysis for Gene Expression Data: A Survey", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, No. 11, November 2004, pp:1370-1386.

[4] Sankar K.Pal, Sanghamitra Bandyopadhyay, and Shubra Sankar Ray, "Evolutionary Computation in Bioinformatics: A Review", *IEEE: Applications and Reviews*, Vol.36, No.5, September 2006, pp:601-615.

[5] Abdelghani Bellaachia, David Portnoy, Yidong Chen, and Abdel. G. Elkahloun, "E- CAST: A Data Mining Algorithm for Gene Expression Data", *Workshop on Data Mining in Bioinformatics* (with SIGKDD02 conference), Vol. 2, 2002, page:49-54.

[6] Dongsong Zhang and Lina Zhou "Discovering Golden Nuggets: Data Mining in Fi- nancial Application", *IEEE Transactions on systems, man, and cybernetics-part c: applications and reviews*, Vol. 34, no.4, November 2004, pp:513-522.

[7] Sharan R, Elkon R, and Shamir R, "Cluster Analysis and its Applications to Gene Expression Data", *Ernest Schering workshop on Bioinformatics and Genome Analysis*, Springer Verlog, 2002.

[8] Dr. Diego Kuonen, \Challenges in bioinformatics for statistical data miners.",*Bulletin of the Swiss Statistical Society*, Vol.-46, October 2003, pp:10-17.

[9] Darlene R. Goldstein, Debashis Ghosh and Erin M. Conlon, "Statistical Isuues in the Clustering of Gene Expression Data", *Statistica Sinica*, Vol 12, 2002,pp:219-240.

[10] Gregory Piatetsky-Shapiro and Pablo Tamayo, "Microarray Data Mining: Facing the Challenges," *SIGKDD Explorations*,Vol. 5, 2003, No. 2, pp:1-5.

[11] Houle, J.L., Cadigan, W., Henry, S., Pinnamaneni, A. and Lundahl, S. "Database Mining in the Human Genome Initiative". Whitepaper, Biodatabases.com, Amita Cor- poration, March 2004.

[12] Herrero J., Valencia A. and Dopazo J, “A hierarchical unsupervised growing neural network for clustering gene expression patterns", *Bioinformatics*, Vol:17, 2001, pp: 126-136

[13] Eisen, Michael B., Spellman, Paul T., Brown, Patrick O. and Botstein, David . “Clus- ter analysis and display of genome-wide expression patterns". Proc. *Natl. Acad. Sci. USA*, Vol.95, No.25, December 1998, pp:14863-14868.

[14] Haux, R., and Kulikowski,C. “Digital Libraries and Medicine", *IMIA Yearbook of Medical Informatics*, Vol. 2, 2001 , pp: 83-99.

[15] Brazma, Alvis and Vilo, Jaak. “Minireview: Gene expression data analysis,"*Federa- tion of European Biochemical societies*, Vol. 480, June 2000, pp:17-24.

[16] Heyer LJ., Kruglyak S., Yooseph S.expression data: identification and analysis of coexpressed genes", *Genome Res*, Vol. 9, No. 11, 1999, pp:11061115.

[17] Jain, A.K., Murty, M.N. and Flynn, P.J. “Data clustering: a review". *ACM Computing Surveys*, Vol. 31, September 1999, No.3, pp: 254-323.

[18] Kjersti Aas, “Microarray Data Mining: A Survey", *Norwegian Regnesentral Comput- ing Center*, vol. 23, February 2001, pp:114-149.

[19] Tamayo P., Solni D., Mesirov J., Zhu Q., Kitareewan S., Dmitrovsky E., Lander E.S. and Golub T.R. “Interpreting patterns of gene expression with self-organizingmaps: Methods and application to hematopoietic differentiation". *Proceedings of National Academy of Science USA*, Vol. 96, No. 6, March 1999, pp: 2907-2912.

[20] Drowning in data. The Economist 1999 (26 June 1999).

[21] P.Hancen and B.Jaumard, “Cluster analysis and mathematical programming",*Math- ematical programming*, Vol. 79, 1997, pp: 191-215.

[22] Judice L.Y.Koh1, Mong Li Lee, Asif M. Khan, Paul T.J. Tan1 and Vladimir Bru- sic,”Duplicate Detection in Biological Data using Association Rule Mining",*Proceed- ings of the Second European Workshop on Data Mining and Text Mining in Bioinfor- matics*, 2004, pp:35-41.

[23] Jinyan Li and Hwee-Leng Ong, “Feature Space Transformation for Better Under- standing Biological and Medical Classifications", *Journal of Research and Practice in Information Technology*,Vol. 36, No. 3, August 2004.

[24] Hideya Kawaji, Yosuke Yamaguchi, Hideo Matsuda, and Akihiro Hashimoto, “AGraph-Based Clustering Method for a Large Set of Sequences Using a Graph Par- titioning Algorithm", *Genome Informatics*, Vol. 12, 2001, pp:93-102.

[25] Alex A. Freitas, “A Survey of Evolutionary Algorithms for Data Mining and Knowl- edge Discovery", *advances in Evolutionary Computation*, A. Ghosh and S.S. Tsutsui, Ed. New York: Springer-Verlag, 2001.