Clustering of Gene Expression Data

A dissertation submitted in partial fulfillment of the requirements for the M.Tech (Computer Science) degree of the Indian Statistical Institute, Kolkata.

By

V. Venkatraman

under the supervision of

Dr. Rajat K. De
Assistant Professor

Dr. Sanghamitra Bandophadyay
Assistant Professor

Machine Intelligence Unit

INDIAN STATISTICAL INSTITUTE
203, B.T. Road, Kolkata - 700035
21 July, 2004

Indian Statistical Institute,
203, Barrackpore Trunk Road,
Kolkata - 700 035

Certificate Of Approval

This is to certify that this thesis titled "*Clustering of Gene Expression Data*" submitted by V.Venkatraman towards partial fulfillment of requirements for the degree of **M. Tech.** in *Computer Science* at *Indian Statistical Institute, Kolkata* embodies the work done under our supervision.

Rajat Kumar De

Dr. Rajat K. De, 16.07.04
Machine Intelligence Unit,
Indian Statistical Institute,
Kolkata - 700 108.

(External Expert)

Dr. Sanghamitra Bandyopadhyay,
Machine Intelligence Unit,
Indian Statistical Institute,
Kolkata - 700 108.

Acknowledgement

I take pleasure in thanking Dr. Rajat K. De and Dr. Sanghamitra Bandyophadyay for their friendly guidance throughout the dissertation period. Their pleasant and encouraging words have always kept my spirits up.

Finally I take the opportunity to thank my classmates for their encouragement to finish this work.

Venkatrama

V. Venkatraman

Abstract

In this thesis we review some standard clustering algorithms and use them to analyze the gene expression data. We also improve upon one of these algorithms which leads to better results on certain data sets. We also discuss a case based system to select prototypes in the data set and apply the clustering algorithms upon the resultant prototypes. This approach results in reduced time complexity of the clustering algorithms while maintaining the quality of the clusters obtained on the original data sets. The results of the algorithms are presented on the breast cancer data set, yeast data set and a simulated data set.

Contents

Chapter 1

Introduction

1.1 Gene Expression

In each and every organism, different genes are active in different cells/tissue types and the level of this activity changes under different conditions. These conditions could be stages of a cell cycle, environmental conditions, diseases etc. The measure of this activity level is called gene expression. Analysis of the these variations in the activity levels can lead to a better understanding of diseases and in the development of drugs to treat those diseases.

One of the most important task of the cell is protein synthesis [5]. Each protein has a specific function. When the cell needs a particular protein, the nucleotide sequence of the appropriate portion of the long DNA molecule in a chromosome is first copied into RNA (through a process called *transcription*). It is these RNA copies of segments of the DNA that are used directly as templates to direct synthesis of the protein (through a process called *translation*). The flow of genetic information in cells is therefore from DNA to RNA to protein. So in the above process the decoding of the genome (information in an organism's DNA) takes place to produce protein.

The first step in decoding a genome is the process of transcription by which an RNA molecule is produced from the DNA of a gene. Transcription and translation are the means by which cells read out, or express, the genetic instructions in their genes. Each gene can be transcribed and translated with a different efficiency, allowing the cell to make vast quantities of some proteins and tiny quantities of others. The amount of a protein depends on the corresponding amount of RNA produced from gene; more is the amount of RNA more will be the protein. Thus this amount of RNA produced from gene is considered as a measure of expression/activity level of the gene. Also cells can control (regulate) the expression of each of its genes according to the needs.

1.2 Microarray Technology

The study of gene expression has been greatly facilitated by the development of microarray technology. There are mainly two types of microarrays, viz., cDNA microarray developed in Stanford University and oligonucleotide microarray developed by Affymetrix Corporation. Here we describe cDNA microarray, in brief, as we are considering it in our experimentation. cDNA microarrays allow us to study genome-wide patterns of gene expression in any given cell type, at any given time, and under any given set of conditions [1]. cDNA microarrays consist of thousands of individual DNA sequences printed in a high density array on a glass microscope slide using a robotic arrayer. The relative abundance of these spotted DNA sequences in two DNA or RNA samples may be assessed by monitoring the differential hybridization of the two samples to the sequences on the array. For mRNA samples, the two samples are reverse-transcribed into cDNA, labeled using different fluorescent dyes mixed (red-fluorescent dye Cy5 and green-fluorescent dye Cy3). After the hybridization of these samples with the arrayed DNA probes, the slides are imaged using scanner that makes fluorescence measurements for each dye. The log ratio between two intensities of each dye is used as gene expression data.

Chapter 2 discusses control of gene expressions. Chapter 3 describes the task of clustering and the some clustering algorithms. In Chapter 4, we provide a modification to the DIANA algorithm and a case based system for case selection prior to clustering. Chapter 5 deals with the description of both simulated and real life data set used in analysis. Chapter 6 analyzes the results obtained with the data sets, while concluding remarks are mentioned in Chapter 7.

Chapter 2

Control of Gene Expression

2.1 Controlling Factor

The cell types in a multicellular organism become different from one another because they synthesize and accumulate different sets of RNA and protein molecules. The different types of cells arise because different sets of genes are expressed. Also the cells can change the pattern of gene expression in response to changes in their environment, such as signals from other cells. Although all of the steps involved in expressing a gene can in principle be regulated, for most genes the initiation of RNA transcription is the most important point of control. The differential expression of genes is also evident in the development cycle of cells and in the difference between the diseased cell types and normal cell types.

The expression level of a gene may be regulated in various ways. Some of these are given.

1. Controlling when and how often a given gene is transcribed (transcription control). The transcription of each gene is controlled by a regulatory region of DNA relatively near the site where transcription begins.

2. Controlling how the RNA transcript is spliced or otherwise processed (RNA processing control). The processes by which gene expression can be controlled at this stage include attenuation of RNA transcript by its premature termination, alternative splice-site selection and RNA editing.

3. Selecting which completed mRNAs in the cell nucleus are exported to the cytosol and determining where in cytosol they are localized (RNA transport and localization control). It has been observed that only a fraction of the RNA synthesized ever leaves the nucleus and only completely processed RNA molecules are sent out of the nucleus.

4. Selecting which mRNAs in the cytoplasm are translated by ribosomes (translational control).

5. Selectively destabilizing certain mRNA molecules in the cytoplasm (mRNA degradation control). Gene expression can be controlled by a change in mRNA stability. The unstable mRNAs code for regulatory proteins, such as growth factors and gene regulatory proteins, whose production rates need to change rapidly in cells. The stability of mRNA can also be changed in response to extracellular signals.

6. Selectively activating, inactivating, degrading, or compartmentalizing specific protein molecules after they have been made (protein activity control).

2.2 Gene Expression Analysis

Gene expression analysis involves looking for informative patterns in multidimensional data obtained from microarray experiments. The main purpose of gene expression analysis could be any of the following

- Identifying function of new genes.

- Discovering new therapeutic drug targets.

- Studying diseases like cancer where the goal is to identify the expression differences responsible for the change from normal to cancerous cells.

- Grouping genes with correlated expression profiles or finding groups of genes participating in the same biological process.

- Studying transcriptional change in response to environmental stimulus. For example to identify which genes control the response and which are affected by it.

- Interpreting gene expression in terms of metabolic pathways which is done in the case of yeast data set mentioned above [4].

- Identifying genetic regulatory networks conveying the interactions between genes and proteins.

Gene expression analysis that we discuss in this thesis mainly relates to clustering genes by expression patterns to look for correlated gene expression patterns. Also we discuss the use of clustering techniques to identify the expression differences responsible for the change from normal to diseased cells.

Chapter 3

Clustering

The term clustering refers to the grouping of a set of data points based on their properties so as to achieve maximum intra-cluster similarity and maximum inter-class dissimilarity. As opposed to classification, clustering is unsupervised.

For the task of clustering we, first of all, need to measure similarity between a pair of data points. This is followed by grouping these data points based on the similarity values. The similarity among the data points can be defined using measures like Euclidean distance, correlation coefficient etc. We use Euclidean distance as a measure of similarity in our analysis of gene expression data. The clustering algorithms are used to estimate the distribution of the data, analyze the clusters and also as a forerunner to classification.

The results, when clustering algorithm is applied to a sample data set, can give two different interpretations.

1. When the genes column is considered as the number of data points and the number of samples is considered as the number of dimensions, then the clustering results can be used to predict which genes are co-regulated.

2. The case in which the number of genes is considered as the number of dimensions and number of samples is considered as the number of data points is usually useful in the case of samples being from different disease types. In this case the clustering results may indicate that the different disease type samples are clustered separately. In the case of time series data set the clustering is done to predict the co-regulated genes as discussed above.

Here we describe some standard clustering algorithms, that we have taken into consideration in our experimentation. These include :

1. K-means algorithm.

2. Agglomerative hierarchical Clustering Algorithms like single linkage, complete linkage, average linkage algorithm.

3. K-ary Clustering algorithm.

4. Diana Clustering algorithm.

3.1 K-means

K-means is a well known partition based clustering algorithm [9]. In this algorithm, each cluster is represented by its mean. It requires user to specify number of clusters in advance. Given k, the algorithm first randomly picks up k data objects and assigns them as cluster means. It then (re)allocates the remaining data objects to that cluster to which the object is closest. The cluster mean is then recomputed for each cluster. This process continues till the membership of the objects becomes stable. The detailed algorithm is given below.

1. Choose, randomly, k points in the data set D and initialize them as a set of mean points M.

2. For each point, $p \in D$ do (3),

3. Find $b \in M$, such that $d(p, b) \geq d(p, m), \forall m \in M$. Assign p to the cluster C_b, where C_b is the cluster with mean b.

4. Let $M' = M$. Reconstruct set M by finding mean of clusters C_1 to C_k and adding them to set M.

5. If $M' = M$, then stop, else go to (2).

The distance between points is measured using metrics like Euclidean, Manhattan, and Mahanalobis distance. This algorithm has a time complexity of $O(ntk)$, where n is the number of data sets, k is the number of clusters, and t is the number of iterations. The k-means is a fast clustering algorithm as it converges quickly. The disadvantage of the algorithm is that the quality of the clusters produced varies quite sharply with the choice of the initial mean points. For example if two of the initial mean points chosen lie within the same cluster then the result will be poor. Another disadvantage of this algorithm is that the number of clusters expected needs to be given as an input to the algorithm.

3.2 Hierarchical clustering algorithms

In hierarchical clustering the goal is to create a hierarchy or a tree in which nodes represent the clusters. Hierarchical clustering techniques can be divided into agglomerative and divisive methods. Agglomerative methods consider each object to belong to a singleton cluster and proceed by a series of fusions of these n clusters into coarser clusters. Divisive methods proceed by starting with a single cluster and then successively separate the clusters into finer clusters. We now consider three agglomerative methods single linkage, complete linkage and average linkage. These three methods differ in the criterion used to fuse the clusters. The algorithm is given below.

1. Let $D = \{D_1, D_2, ..., D_n\}$. Each data point placed in its own cluster, creating a list of clusters $L = \{L_1, L_2, ..., L_n\}$.
2. Find L_m and L_n, such that $mf(L_m, L_n) \geq mf(L_i, L_j), \forall L_i, L_j \in L$. The function $mf()$ is called the merge cost function.
3. Remove the clusters L_m and L_n from L.
4. Merge the two clusters L_m into L_n a single cluster and add the cluster to L.
5. Go to (2) until there is only one cluster in L.

The merging cost function may have three forms,

$$mf(L_i, L_j) = \min_{x \in L_i, y \in L_j}(x, y) \tag{3.1}$$

$$mf(L_i, L_j) = \max_{x \in L_i, y \in L_j}(x, y) \tag{3.2}$$

$$mf(L_i, L_j) = avg_{x \in L_i, y \in L_j}(x, y) \tag{3.3}$$

and corresponding to above three forms the resulting algorithms are called single linkage, complete linkage and average linkage respectively. The time complexity of all the three methods is $O(n^3)$ as the construction of tree takes n steps and in each step of construction there are $O(n^2)$ comparisons to find the two clusters to merge.

One of the advantages of these methods is that once two data points are assigned to a single cluster there is no way for them to get separated at later stage. These methods, especially single linkage, are susceptible to noise. For example one data point (noise) may lead to fusing of two distinct clusters into a single cluster.

3.3 K-ary Clustering Algorithm

K-ary clustering algorithm is a agglomerative hierarchical clustering technique [2]. This algorithm constructs a k-ary tree, where each internal node can have up to k child nodes. One of the assumptions in this methods is that relying on the similarities among large groups of genes helps reduce the noise effects that are inherent in expression data [12]. k is the upper bound on the number of children of each internal node so it allows us to highlight some actual clusters since nodes with less than k children represent a set of data points that are similar but significantly different from rest of the data points. The detailed algorithm is given below

1. Each data point is placed in its own cluster, creating a list of clusters C.
2. For each j$\in$C, construct L$_j$, the ordered linked list of data points based on the similarity to j. Compute b$_j$ = j $\cup$ first k-1 data points of L$_j$.
3. For i = 1 to (n-1)/(k-1) do,
4. Find b = argmax$_{j\in C}$\{V(b$_j$)\}, $C = C \setminus b$. Let p = min\{m $\in$ b\}.
5. For all clusters j $\in$ C do,
6. S(p,j) = $\sum_{m\in b} S(m,j) \div \sum_{m\in b}$
7. Remove all clusters in b from L$_j$. Insert p into L$_j$.
8. b$_j$ = j $\cup$ first k-1 clusters of L$_j$. Go to (5).
9. C = C $\cup$ p. Generate L$_p$ from all clusters in C and find b$_p$.
10. Go to (3).
11. Return C, where C is a singleton which is the root of the tree.

The complexity of this algorithm is $O(n^3)$, which is same as that of hierarchical clustering algorithms.

3.4 DIANA

DIANA (Divisive Analysis) is a divisive hierarchical clustering technique, first introduced in [10]. Initially (step 0), there is one large cluster consisting of all n data points. At each subsequent step, a cluster is split into two clusters until finally all clusters. comprise of single objects. Thus, the hierarchy is built in n-1 steps. In each iteration, the cluster with the largest diameter is chosen for partitioning. Let us define a cluster C such that $C = p_1, p_2, ..., p_n$, where $p_1, p_2, ..., p_n$ are the data points in the cluster C. Then the diameter D of a cluster is defined as,

$$D = \max_{\forall p_i, p_j \in C} d(p_i, p_j) \tag{3.4}$$

where d is the distance measure between two data points. The cluster is split by first considering a point with the highest average dissimilarity to other points in the cluster. This point initiates a new cluster. Then for each point in the original cluster, we find its average distance from the both the new and original clusters and the point is assigned to the cluster with the least average distance. Detailed algorithm is given below.

1. Find the object, which has the highest average dissimilarity to all other objects. This object initiates a new cluster, called the splinter group.

2. For each object i outside the splinter group compute

3. D_i = [avg d(i,j)$\notin$ $R_{splintergroup}$] - [avg d(i,j)$\in$ $R_{splintergroup}$]

4. Find an object h for which the difference D_h is the largest. If D_h is positive, then h is, on the average close to the splinter group.

5. Repeat steps 2 and 3 until all differences D_h are negative. The data set is then split into two clusters.

6. Select the cluster with the largest diameter. The diameter of a cluster is the largest dissimilarity between any two of its objects. Then divide this cluster, following steps 1-4.

7. Repeat Step until all clusters contain only a single object.

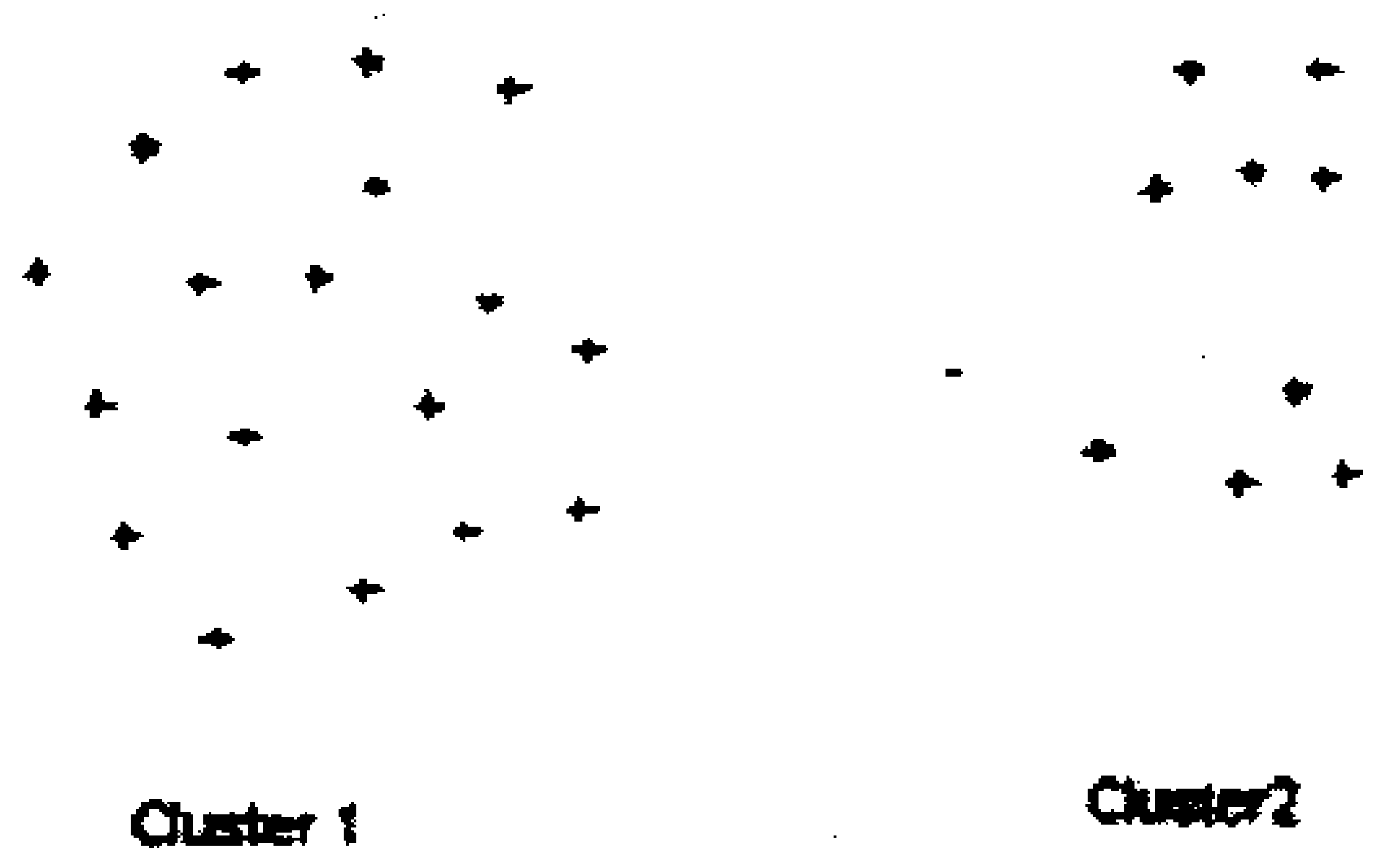
As can be seen in the results section the DIANA algorithm gives good results on all data sets. The disadvantage of the DIANA algorithm is that the time complexity of the is high.

Chapter 4

Modification to DIANA and Selection of Prototypes

4.1 Modification to DIANA

DIANA partitions the clusters based on the criterion of maximum diameter of the cluster. There are cases where DIANA wrongly partitions a large homogeneous cluster rather than a cluster with two small homogeneous clusters inside it. This case is shown in the figure below.



As shown in the above figure, the DIANA will incorrectly partition the large cluster, Cluster 1 rather than Cluster 2, in spite of the fact that Cluster 1 is more dense in comparison to the Cluster 2 which has two distinct clusters contained in it.

To rectify the above said anomaly the partitioning criterion in the standard DIANA algorithm was changed from maximum diameter to maximum density among the clusters. Let the diameter of the cluster C be D and the the number of data points in the cluster be N, then the density d of the cluster C, using equation 3.4, is given by,

$$d = D/N \tag{4.1}$$

Now we will consider the scenario shown in the above figure and see how the modification to DIANA algorithm leads to better result. We can observe that Cluster 1 has higher diameter compared to Cluster 2, but as Cluster 1 has greater number of data points than the Cluster 2, the Cluster 1 has lesser density than Cluster 2. Modified DIANA algorithm would choose Cluster 2 for partitioning rather than the Cluster 1. The modified DIANA algorithm also gives good results with other data sets we have considered in our experimentation, which can be seen in the results section. The computational complexity of the modified DIANA is almost same as that of the original DIANA algorithm, as the diameter of the cluster is calculated in the original DIANA algorithm and the number of points in each cluster is also known to us in original DIANA algorithm. The only extra operation done in each iteration of the modified DIANA algorithm is the calculation of the density using the diameter of the cluster and the number of points in the cluster.

4.2 Selection of Prototypes

A case-based system adapts old solutions to meet new demands, explains and critiques new situations using old instances (called cases), and performs reasoning from precedents to interpret new problems [11]. Case-based system in contrast to traditional knowledge based system, operates through a process of remembering one or a small set of concrete instances or cases and basing decisions on comparisons between the new situation and old one. In this thesis we describe a model for selecting cases in data sets prior to clustering. The clustering algorithms are applied on the selected cases. This model discussed here is derived from [3].

Initially all the points in the data set belong to a single class. Cases are viewed as labeled patterns which represent different regions of the class. A notion of fuzzy similarity, using π-type membership function, is incorporated together with repeated insertion and deletion of cases in order to determine a stable case base. Let $x = [x_1, x_2, ..., x_i, ..., x_n]$ be point in an n-dimensional feature space. $\xi_k = [\xi_{k1}, \xi_{k2}, ..., \xi_{ki}, ..., \xi_{kn}]$ denotes kth case in the case base. $\mu_k(x)$ represents the degree of similarity of x to a case ξ_k. $d_k(x)$ stands for the distance between x and ξ_k. The degree of similarity between a point x and a case ξ_k is defined as

$$\mu(x) = \begin{cases} 1 - 2(\frac{d(x)}{\lambda})^2 & 0 \leq d(x) < \lambda/2 \\ 2[1 - \frac{d(x)}{\lambda}]^2 & \lambda/2 \leq d(x) < \lambda \\ 0 & \text{otherwise} \end{cases} \tag{4.2}$$

where λ is the bandwidth of $\mu_k(x)$, i.e., the separation between its two (crossover) points where $\mu_k(x) = 0.5$. The distance d_k is taken as the Euclidean distance.

Effect of λ: As λ increases, the extent of the region around a case increases, and therefore the number of cases required for representing the data set decreases. This implies that the generalization capability of an individual case increases with increase in λ. Initially, although the number of cases decreases with the increase in λ, the generalization capability of individual cases dominates. For further increase in λ, the number of cases becomes so low that the generalization capability of the individual cases may not cope with the proper representation of class structures. Algorithm for finding cases is described below.

1. Select an initial point x randomly from the data set. Add x to the case base ξ.
2. Choose a point x randomly from the data set. Calculate $\mu_k(x)$, $\forall \xi_k \in \xi$.
3. If $\mu(x) > 0.5$ for at least one case then declare the point to be belonging to the case with maximum $\mu(x)$ value.
4. If $\mu(x) < 0.5$ for all cases in the case base, then declare the data point to be a case and add the case to the case base ξ.
5. Go to (2) till all the points in the data set have been chosen, else go to next step.
6. A case is deleted from the case base for which $\mu(x)$ is minimum and number of data points for which $\mu(x) > 0.5$ is less than some pre-defined value.
7. If the case base has changed from previous iteration then go to (1), else return the case base ξ.

Now we describe how the above algorithm has been used in our analysis. The algorithm was applied on an unclustered data set and the set of cases in the data set were derived as a result. Also we store the data points associated with each case. Then we cluster the cases using any of the clustering algorithms described in the previous chapter. Then in all the clusters we have obtained as the output of clustering algorithm, we substitute in the place of each case, the data points associated with that case. The clusters obtained contain, between them, all the data points in the original data set. The quality of clusters obtained is similar to the clusters obtained when the clustering algorithms are applied to original data set. The advantage of this method that we are clustering data set with reduced number of points, so the clustering algorithm run faster compared to the case when algorithm is run on the original data set. This algorithm is useful in the case, when we need to apply many clustering algorithms to a data set. Then in this case the cost of running the case selection algorithm is offset by the the cost of running the clustering algorithms with the original data set.

Chapter 5

Data Sets

The gene expression data is primarily of two types depending on how the experiments are conducted

- The expression of a set of genes is measured on different sets of samples. The different set of samples generally correspond to different disease types. For example, consider the leukemia data set [6]. This data set has some samples taken from patients with ALL (acute lymphoblastic leukemia) type cancer, some samples taken from patients with AML (acute myeloid leukemia) type cancer and some samples taken from normal persons. The expression of a set of genes are measured in all these samples. These type of data sets are henceforth referred to as samples data sets.

- The expression of a set of genes is measured on single sample over a certain time period. The samples are studied during fixed time intervals. These time intervals usually correspond to some natural processes like cell development cycle. For example in yeast data set [4], the yeast sample is studied during the metabolic shift of the yeast from fermentation to respiration. The sample is studied in an interval of two hours for a total of 14 hours. This type of data set is henceforth referred to as time series data set.

The gene expression data, for the purpose of analysis, is organized as a matrix. Usually the rows represent the genes and the columns represent the samples or the time intervals. The columns and rows can be interchanged depending on the intended result from the gene expression analysis.

Gene expression data are concerned with ratios of the expression levels of two different samples. One of the samples is a test sample and the other is a reference sample. This is done so as to maintain the consistency of the results even if the experiment is repeated under slightly different conditions.

In this chapter we discuss two gene expression data sets and one simulated data sets which we have used in our analysis.

A simulated data set was generated with patterns similar to the gene expression data sets. We explain the methodology used to generate the data. Firstly, we generate 5, 1×20 vectors. Then the discrete cosine transformation of the 5 vectors is computed. The resulting coefficients are again 5, 1×20 vectors. Thus we have 5 data points with dimension 20. We call these data points as seed points. Now considering a seed point, we generate a small increment value, say x, using random normal distribution for each dimension of the seed point.The generated increment values are added to the seed point to get a data point close to the seed point. In this way we generate 500 points around the seed point. The above process is repeated for other seed points. Thus we generate 2500 data points, each with 20 dimensions. These 2500 points roughly fall into 5 clusters with 500 data points each.

The second real life data set is a samples data set. This data set was a result of an experiment to study two major types of breast cancer. Invasive ductal carcinoma(IDC) and invasive lobular carcinoma(ILC) are the two major histological types of breast cancer [7]. The microarray experiments were conducted to study whether IDC and ILC represent molecularly distinct entities and what genes might be involved in the development of these two phenotypes. Total RNA samples from 21 ILCs, 38 IDCs, and 3 normal tissues were amplified and hybridized to 42,000 cDNA microarrays. The data set derived from this experiment was downloaded from Stanford Microarray Database (SMD). The data set has 49 data points (samples) and 42,000 dimensions (genes).

Our first real life data set is a time series data set. In this data set DNA microarrays containing almost every gene of *Saccharomyces cerevisiae* (yeast) were used to measure the gene expression accompanying the metabolic shift from anaerobic (fermentation) to aerobic (respiration) metabolism [4]. This shift is called diauxic shift. This shift from anaerobic growth to aerobic respiration upon the depletion of glucose is correlated with widespread changes in the expression of genes involved in fundamental cellular processes such as carbon metabolism, protein synthesis, and carbohydrate storage. DNA microarrays, containing approximately 6400 distinct DNA sequences, were printed onto glass slides by using robotic printing device. The samples harvested at seven successive 2-hour interval and mRNA was isolated and gene expression measurements were taken. The data set derived from this experiment was downloaded from Stanford Microarray Database (SMD). The data set has 6153 data points (genes) with number of dimensions seven (time intervals).

Chapter 6

Results

Before presenting the results we discuss two cluster validity measures in order to investigate how clustering algorithms can be efficiently applied on the gene expression patterns. The quality of the clusters produced by the clustering algorithms can be measured using two type of scores called Jaccard score and Minkowski Score.

6.1 Jaccard Score

Jaccard Score [8] is given by,

$J(T, S) = \frac{n_{11}}{n_{11}+n_{10}+n_{01}}$,

where T is the true solution we expect from an ideal clustering algorithm and S is the solution obtained from the clustering algorithm we wish to measure. The terms n_{01}, n_{10}, n_{11} are defined below.

- n_{01} - Number of pairs of elements that are in same cluster only in S.
- n_{10} - Number of pairs of elements that are in same cluster only in T.
- n_{11} - Number of pairs of elements that are in same cluster in both S and T.

Intuitively, higher the Jaccard score, better is the clustering.

6.2 Minkowski Score

Minkowski Score [13] is given by,

$M(T.S) = \sqrt{\frac{n_{01}+n_{10}}{n_{11}+n_{10}}}$

The meaning of the symbols is the same as for the case of Jaccard Score. Lower the Minkowski Score, better is the clustering.

6.3 Clustering

The clustering algorithms, described in Chapter 4, were run on the simulated data set with 2500 data points, where each data point had a dimension of 20. The clustering algorithms were run with expected number of clusters equal to 5. The Minkowski and Jaccard Scores for the clusters produced are given in Table 6.1.

Table 6.1: Results with Simulated Data Set.

	Jaccard Score	Minkowski Score
k-means	0.7	0.419
k-ary	0.74	0.395
Average Linkage	0.71	0.401
DIANA	0.735	0.4
M-DIANA	0.754	0.3804

As we can observe from the Table 6.1, the Jaccard score of the M-DIANA algorithm is maximum and Jaccard score of the k-means algorithm is minimum among all the clustering algorithms, when applied on the simulated data set. This implies that the M-DIANA gives best clustering result and the k-means algorithm gives worst clustering result on the simulated data set. This result is verified by observing the Minkowski score for the clustering algorithms on the simulated data set. The Minkowski score of the M-DIANA algorithm is minimum and the score of the k-means algorithm is maximum.

The clustering algorithms were run on the breast cancer data set to differentiate between the the samples corresponding to different histological types of breast cancer, e.g, invasive ductal carcinoma (IDC) and invasive lobular carcinoma (ILC). The data set has 49 data points, where each point has dimension of 42,000. So the clustering algorithms were run with expected number of clusters equal to two. The Minkowski and Jaccard Scores for the clusters produced are given in Table 6.2.

As we can observe from the Table 6.2, the Jaccard score of the M-DIANA algorithm is maximum and Jaccard score of the average linkage algorithm is minimum among all the clustering algorithms, when applied on the breast cancer data set. This implies that the M-DIANA gives best clustering result and the average linkage algorithm gives worst clustering result. This result is verified by observing the Minkowski score for the clustering algorithms on the breast cancer data set. The Minkowski

Table 6.2: Results with Breast cancer data set.

	Jaccard Score	Minkowski Score
k-means	0.63	0.557
k-ary	0.683	0.552
Average Linkage	0.63	0.557
DIANA	0.6501	0.561
M-DIANA	0.7	0.546

score of the M-DIANA algorithm is minimum and the score of the average linkage algorithm is maximum.

We can observe the variance in the clustering results depending on the data sets on which the clustering algorithms are applied. For example, in the case of simulated data set the average linkage algorithm gives better results compared to k-means algorithm, whereas in the case of breast cancer data set the k-means algorithm gives better results compared to average linkage algorithm.

In the results we have also observed that 75 genes show high expression in ILCs amd low expression in IDCs, and 75 genes vice versa. Most of these 150 genes can be categorized into five biological processes: cell adhesion/mobility, lipid/fatty acid metabolism, immune and defense response, electron transport and nucleosome assembly [7].

The clustering algorithms were applied on the yeast data set. The yeast data set being time series data set the goal was to check whether the functionally related or co-regulated genes are clustered together, as this could provide clues to the functionality of genes about which we do not have information.

In results using all the clustering algorithms we have observed that genes MLS1, IDP2, ICL1, ACS1, ACR1, FBP1 and PPC1 are clustered together. These seven genes were known to be glucose repressed and five of these seven (MLS1, ICL1, ACS1, ACR1 and FBP1) genes were noted to share common upstream activating sequence (UAS), the carbon source response element (CSRE) [4]. The set of seven other genes (GSY1, CTT1, HSP42, HSP26, HSP12, YKL026C and YGR043C) forms a separate cluster. All the seven genes contain stress response elements (STRE), and with the exception of HSP42 have been previously found to be controlled at least in part by these STRE [4].

We now consider the clusters produced by the M-DIANA algorithm with the simulated data set and breast cancer data set and compare these clusters with the clusters produced by other clustering algorithms discussed in Chapter 4. The comparison for the simulated data set is given in Table 6.3 and for the breast cancer data set is given in Table 6.4.

Table 6.3: Cluster Comparison for Simulated data set.

	Cluster 1 (489)	Cluster 2 (506)	Cluster 3 (495)	Cluster 4 (502)	Cluster 5 (508)
k-means	9	4	2	5	6
k-ary	3	3	0	2	1
Average Linkage	4	0	8	9	2
DIANA	2	2	1	3	2

We now explain the Table 6.3. The first row of the table gives the clusters produced by the M-DIANA algorithm, where the values in the paranthesis give the number of data points in the corresponding cluster. For each subsequent row, the clusters produced by the corresponding algorithm are compared with the clusters produced by the M-DIANA algorithm. For example the value in the first row, third coloumn gives the number of data points not present in the Cluster 2 of the clusters produced by the k-means algorithm, that were present in the Cluster 2 of the clusters produced by the M-DIANA algorithm. Consider the clusters produced by the k-means algorithm. The cluster which has maximum number of data points common with the Cluster 1 of the clusters produced by the M-DIANA algorithm, is assigned as Cluster 1. This process is repeated for all the clusters. This scheme for assigning the cluster names is used with all other clustering algorithms.

From Table 6.3 we can infer that clusters produced by the k-ary algorithm are most similar to the clusters produced by the M-DIANA algorithm when applied on the simulated data set. Also the clusters produced by DIANA algorithm are also similar to the clusters produced by the M-DIANA algorithm.

The comparisons for the breast cancer data set is given in Table 6.4. Table 6.4 has the same format as that of the Table 6.3.

In the Table 6.4, we observe that the clusters produced by all the clustering algorithms are similar to the clusters produced by the M-DIANA algorithm, but this may be a misleading result as the number of data points in the breast cancer data set is small (49) and thus may result in fewer differences between the clusters produced.

6.3.1 Case based system

The case based system for selecting prototypes was applied on the simulated data set. The data set had 2500 points and we took the parameter $\lambda = 0.75$. The algo-

Table 6.4: Cluster Comparison for Breast cancer data set.

	Cluster 1 (33)	Cluster 2 (26)
k-means	1	0
k-ary	1	0
Average Linkage	1	0
DIANA	0	1

rithm returned 804 resultant prototype data points. Then the prototype points were clustered using all the clustering algorithms discussed in Chapter 3. This process was discussed in Chapter 4. Now for the purpose of analysis, we consider the clusters produced by running each clustering algorithm on prototype points as the proposed solution S and the the solution returned by the clustering algorithm on the simulated data set as the true solution T. Then we calculate the Jaccard and Minkowski score in the case of each algorithm. The result is given in Table 6.5.

Table 6.5: Prototyping results for simulated data set.

	Jaccard Score	Minkowski Score
k-means	0.91	0.21
k-ary	0.907	0.213
Average Linkage	0.914	0.209
DIANA	0.926	0.2
M-DIANA	0.921	0.205

The high Jaccard Scores and low Minkowski scores for all the clustering algorithms indicates that the prototype points returned by the case based system represent in good measure the original simulated data set for the purpose of clustering.

The case based system for selecting prototypes was also applied on the breast cancer data set. The data set had 49 points, where each data point has dimension 42,000 and we took the parameter $\lambda = 13$. The algorithm returned 17 resultant prototype data points. This result was also analysed the way it was done for simulated data as explained above. The results are shown in Table 6.6.

In the results for the breast cancer data set also we observe the same trend we observed in results for simulated data set. Here also the resultant prototype points represent the original breast cancer data set for the purpose of clustering, but to a lesser extent as compared to simulated data set. This can be inferred from the relatively low Jaccard scores in the results for breast cancer data set as compared to the results for the simulated data set.

Table 6.6: Prototyping results for breast cancer data set.

	Jaccard Score	Minkowski Score
k-means	0.89	0.24
k-ary	0.902	0.22
Average Linkage	0.89	0.24
DIANA	0.896	0.226
M-DIANA	0.91	0.218

6.4 Timing Comparison

In this thesis we also present the relative comparison of the time requirments of different clustering algorithms. The results are shown with simulated data set and the breast cancer data set. The algorithms were run on the simulated data set, with the expected number of clusters $k = 5$ and on the breast cancer data set, with the expected number of clusters $k = 2$. The results are shown in table given below.

Table 6.7: Timing Comparison.

	Simulated Data Set (secs)	Breast Cancer Data Set (secs)
k-means	206,639	38
k-ary	394	94
Average Linkage	371	69
DIANA	357	42
M-DIANA	361	49
Case-Based System	423	72

Two values corresponding to k-means clustering algorithm show time taken with two different initial set of mean points. Here we observe that the there is sharp variance in the time for convergence of the algorithm depending on the choice of initial mean points. As can be observed in the Table 6.6, k-means algorithm has the least running times and the k-ary clustering algorithm has the highest running time, respectively, among the clustering algorithms discussed. The time complexity of the k-means algorithm is $O(ntk)$, where n is the the number of data points, t is the number of iterations and k is the number of clusters. The time complexity of the k-ary algorithm and all other heirarchical algorithms is of the order $O(n^3)$. In both the data sets the number of expected clusters is far less than the the number of data points. Thus the running time of the k-means is lowest whereas the running time of the k-ary algorithm is highest.

The case based system for selecting prototypes was applied on the simulated data set. The data set had 2500 points and we took the parameter $\lambda = 0.75$. The time taken to run this algorithm was 423 seconds. The prototype points were clustered using the clustering algorithms. The case based system for selecting prototypes was also applied on the breast cancer data set. The data set had 49 points and we took the parameter $\lambda = 13$. Thetime taken to run this algorithm was 72 seconds.

Chapter 7

Conclusions and Discussions

In this thesis, we have analyzed the gene expression data using some clustering techniques. We also discussed a modification to the DIANA algorithm where the clustering criterion was changed from maximum diameter to maximum density. As we have seen in the results this algorithm gives better clusters than the standard DIANA algorithm, particularly with the real life gene expression data sets. The time complexity of the modified algorithm remains the same.

We also had discussed a case based system to select prototypes from a data set and applied clustering techniques on the resultant prototype points. This approach as we have seen in the results leads to faster clustering whereas retaining the cluster quality predicted on original data set.

Bibliography

[1] *Nature Genetic Supplement*, 21, January, 1999.

[2] Z. Bar-Joseph, E.D. Demaine, and D.K. Gifford. K-ary clustering with optimal leaf ordering for gene expression data. *Bioinformatics*, 19:1070–1078, 2003.

[3] Rajat K. De and Sankar K. Pal. A connectionist model for selection of cases. *Information Sciences*, 132:179–194, 2001.

[4] Joseph L. DeRisi, Vishwanath R. Iyer, and Patrick O. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 278:680–686, 1997.

[5] Bruce Alberts et al. *Molecular Biology of the Cell*. Garland Science, New York, 4 edition.

[6] Golub et al. Molecular classification of cancer: Class discovery and class prediction by gene expression. *Science*, 286:531–537, 1999.

[7] Zhao et al. Different gene expression patterns in invasive lobular and ductal carcinomas of the breast. *Mol. Biol. Cell.*, 15:2523–2536, 2004.

[8] B. Everitt. *Cluster Analysis*. Edward Ronald, London, 3 edition.

[9] E. Forgy. Cluster analysis of multivariate analysis: Efficiency vs interpretability of classifications. *Biometrics*, 21:168, 1965.

[10] L. Kauffman and P.J Rousseeuw. *Fitting Groups in Data. An Introduction to Cluster Analysis*. Wiley, New York, 1990.

[11] J.L. Kolodner. *Case-Based Reasoning*. Morgan Kauffman, San Mateo, 1993.

[12] R. Elkon R. Sharan and R. Shamir. Cluster analysis and its applications to gene expression data. *Ernst Schering workshop on Bioinformatics and Genome Analysis*, 2001.

[13] R.R. Sokal. Clustering and classification: background and current directions. *Classification and Clustering*, pages 1–15, 1977.