Abstract - The advent of DNA microarray technology has enabled biologists to monitor the expression levels (MRNA) of thousands of genes simultaneously. In this survey, we address various approaches to gene expression data analysis using clustering techniques. We discuss the performance of various existing clustering algorithms under each of these approaches. Proximity measure plays an important role in making a clustering technique effective. Therefore, we briefly discuss various proximity measures. Finally, since evaluation of the effectiveness of the clustering techniques over gene data requires validity measures and data sources for numeric data, we discuss them as well.

Keywords - Gene expression data, proximity measure, coherent pattern, clustering, cluster validation

I. INTRODUCTION

Microarray technology has made it possible to simultaneously monitor expression levels of many thousands of genes through two major types of microarray experiments: cDNA microarray [1] and oligonucleotide arrays [2]. Although both types of experiments follow different protocols, they have some common basic procedures. The analysis of gene expression is shown in Fig. 1 at a very high level.

Fig 1: Gene expression analysis pipeline

The basic steps executed in the process of gene expression data generation are as follows.

- **Chip manufacture**: A microarray is a small chip consisting of a solid surface, onto which tens of thousands of DNA molecules (probes) are attached in fixed grids. Each grid cell corresponds to a DNA sequence. Since thousands of DNA molecules are bonded to a single array, it is possible to measure the expression of many thousands of genes in parallel.

- **Sample preparation, labeling and hybridization**: Two mRNA samples (a test sample and a control sample) are reverse-transcribed into cDNA (targets), labeled using either fluorescent dyes or radioactive isotopes and then hybridized with the probes on the surface of the chip.

- **Washing**: The slides are then washed to remove excess hybridization solution from the array so that only the DNA complementary to each of the features will remain bound to the features on the array.

- **Image Acquisition**: The final step of the process is to produce an image of the surface of the hybridized array where chips are scanned to read the signal intensity that is emitted.

The image of the microarray generated by the scanner is the raw data. Feature extraction software converts the image into numerical information that quantifies gene expression.

A. Pre-processing of gene expression data

A microarray experiment assesses a large number of DNA sequences under multiple conditions. These conditions may be a time series during a biological process or a collection of different tissue samples. Without making any distinction among the DNA sequences in this survey, we will refer to them as genes. Similarly, we will refer to all kinds of experimental conditions as samples. A gene expression dataset from a microarray experiment may be considered an \( n \times l \) matrix \( M \) (as shown in Fig. 2), where \( M = [m_{ij}] \) are the rows which represent expression patterns of a set of \( n \) genes \( (G = \{g_1, g_2, \ldots, g_n\}) \), the columns represent expression profiles of a set of \( l \) samples \( S = (s_1, s_2, \ldots, s_l) \) at different conditions/time points, and each cell \( m_{ij} \) is the expression level of gene \( g_i \) on sample \( s_j \) (\( 1 \leq i \leq n \) and \( 1 \leq j \leq l \)).

Fig 2: A gene expression matrix

The microarray data thus generated is then cleaned, transformed and normalized to resolve any errors, noise and bias introduced by the microarray experiments.

B. Coherent, Co-expressed and Co-regulated Genes

Two genes are said to co-express when they have similar expression patterns and may belong to the same or similar functional categories, whereas two genes are said to co-regulate when they are regulated by common transcription factors. Co-expressed genes in the same group are involved in the same cellular process. A strong expression pattern correlation between those genes indicates co-regulation. A coherent gene expression pattern (or coherent pattern in short) characterizes the common trend of expression levels for a group of co-expressed genes. A coherent pattern is a “template”, while the expression profiles of the corresponding co-expressed genes match the pattern with small divergence. Coherent gene expression patterns may characterize important cellular processes and may also be involved in the regulating mechanism in the cells. Clustering
techniques have proven to be useful in finding coherent or co-expressed gene patterns.

However, as stated in [3], co-regulated genes have very high chance of having similar functions whereas co-expressed genes are not necessarily co-regulated and may not have similar functions.

C. Extraction of Interesting Patterns from Gene Expression Data Using Clustering Techniques

Clustering is the process of grouping data objects into a set of disjoint classes, called clusters, so that objects within a 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. As stated in [4], the various clustering approaches are partitional, hierarchical, density based, model based, grid based and soft computing. In this survey, we will consider the use of these algorithms for gene-based clustering.

D. Proximity Measures

There are different methods [5] for quantifying similarity or dissimilarity among gene expression levels, described in terms of the distance between them in the high-dimensional space of gene expression measurements. A dissimilarity measure \(d_{ij}\) for any two genes \(g_i\) and \(g_j\) obeys the following properties.

i. The distance between any two profiles cannot be negative.

ii. The distance between a profile and itself must be zero.

iii. A zero distance between two profiles implies that the profiles are identical.

iv. The distance between profile \(g_i\) and profile \(g_j\) is the same as distance between profile \(g_j\) and profile \(g_i\), i.e., \(D(g_i, g_j) = D(g_j, g_i)\).

Different proximity measures: A microarray experiment compares genes from an organism under different development time points, conditions or treatments. For an \(n\) condition experiment, a single gene has an \(n\)-dimensional observation vector known as its gene expression profile. A proximity measure is a real-valued function that assigns a positive real number as a similarity value between any two expression vectors. Therefore, to identify genes or samples that have similar expression profiles, selection of an appropriate proximity measure is very essential.

Some of the commonly used proximity measures can be found in [6]. Selection of a measure depends upon the type of data, its dimensionality and the approach used in identification of coherent patterns.

E. Cluster Validity Measures

Cluster validity depends upon (i) the homogeneity in individual clusters, (ii) given the “ground truth” (available from the domain knowledge) of the clusters, and (iii) the reliability of the clusters. A few validation measures commonly used in evaluating the effectiveness of the technique(s) used in gene expression data mining will be highlighted in this survey.

F. Paper Organization

The rest of the survey is organized as follows: Section II describes the formulation of the problem and its complexity. In Section III we discuss proximity measures for numeric data. Various generalized clustering approaches are reported in Section IV. In Section V, cluster validity measures are discussed and Section VI discusses challenges and research issues. Section VII takes a brief look at some gene expression datasets to which gene-based clustering approaches have commonly been applied. Finally, we conclude in Section VIII.

II. PROBLEM FORMULATION

Gene expression data are generated by DNA chips and other microarray techniques and they are often presented as matrices of expression levels of genes under different conditions (including environments, individuals, and tissues). One of the major objectives of gene expression data analysis is to identify groups of genes having similar expression patterns over the full space or subspace of conditions. It may result in the discovery of regulatory patterns or condition similarities. Generally co-expressed genes, which are members of the same clusters, are expected to have similar functions.

Most researchers address this issue by: (i) using either partitional, hierarchical, density-based, model-based or subspace clustering algorithms, based on the proximity between genes or conditions in the expression matrix or (ii) giving equal weights to all conditions or all genes in the computation of gene similarity and vice versa. However, if the proximity measure is not properly selected, it may lead to the discovery of some similar groups of genes at the expense of obscuring other similar groups. Hence, it may be necessary to recover information lost due to over-simplification of similarity and grouping computation, which may reveal the involvement of a gene or a condition in more than one group.

III. EXISTING PROXIMITY MEASURES FOR NUMERIC DATA

Distance or similarity measures are essential to solve many pattern recognition problems such as classification, clustering and retrieval problems. From the scientific and mathematical points of view, distance is defined as a quantitative degree of how far apart two objects are [3]. A synonym for distance is dissimilarity. Distance measures satisfying the metric properties are simply called metric while other non-metric distance measures are occasionally called divergence. Synonyms for similarity include proximity and similarity measures are often called similarity coefficients. The choice of the proximity measure depends upon the type of data, dimensionality and the approach used in the identification of coherent patterns. We reproduce from [6] a partial list of the formula of a few proximity measures for numeric data, since gene expression data are mostly available as numeric data.
In this section, we discuss the problem of clustering genes based on their expression patterns. The purpose of gene-based clustering is to group together co-expressed genes which indicate co-function and co-regulation. We will first review a series of clustering algorithms belonging to various approaches (as reported in Section I-C), which have been applied to group genes. Finally, we present the issues involved in gene-based clustering.

A. Partitioning Approach

Clustering techniques belonging to this approach are divided into two major sub-categories: centroid based where each cluster is represented by using the gravity centre of the instances and medoid based, where each cluster is represented by means of the instances closest to the gravity centre.

K-means [7] is a simple and fast centroid-based clustering algorithm which divides the data into pre-defined number of clusters in order to optimize a predefined criterion. However, it may not yield the same result with each run of the algorithm. To detect the optimal number of clusters, users usually run the algorithm repeatedly with different values of k and compare the clustering results. For a large gene expression dataset, which contains thousands of genes, this extensive parameter fine-tuning process may not be practical. Moreover, gene expression data typically contain a huge amount of noise and the k-means algorithm forces each gene into a single cluster, which may lead to the generation of non-qualitative or biologically non-relevant clusters. It is not suitable to detect clusters of arbitrary shapes.

In spite of its unsuitability, the k-means algorithm is frequently used to cluster gene expression data due to its simplicity and it provides baseline results to compare with when we develop new clustering algorithms. The application of k-means algorithm and its variants to cluster gene expression data are widely used, as is evidenced by [50]-[59], [61] and [62].

k-modes [8] is a faster extension of k-means algorithm to handle categorical data that uses a different similarity measure, replaces k-means with k-modes and uses a frequency based method to update modes. The k-modes algorithm is useful only after the conversion of the numeric data into categorical. However, it may lead to information loss and hence may deteriorate the cluster result.

The working of the Fuzzy C-Means (FCM) [9] algorithm is very similar to k-means. In FCM, each point has a degree of membership to each cluster rather than belonging completely to just one cluster. Thus, points on the edge of a cluster are assigned partial values and an iterative optimization approach to determine the optimal number of clusters is adopted. The algorithms PAM and CLARA [10]. PAM uses dissimilarity values and an iterative optimization approach to determine a representative object for each cluster called medoid. Once the medoids have been selected, each non-selected object is grouped with the medoid to which it is most similar. Otherwise, it declares that a new cluster, to a lesser degree than points in the center of the cluster. Fuzzy C-Means and its variants are widely used techniques for microarray data clustering and they can be found in [55], [60], and [63].

Two early versions of k-medoid methods are the algorithms PAM and CLARA [10]. PAM uses dissimilarity values and an iterative optimization approach to determine a representative object for each cluster called medoid. Once the medoids have been selected, each non-selected object is grouped with the medoid with which it is most similar. Finally, the quality of a clustering is measured by the average dissimilarity between an object and the medoid of its cluster. CLARA follows the same principle as PAM, but instead of finding representative objects for the entire dataset, it draws a sample of the dataset, and applies PAM to this sample. It then classifies the remaining objects using partitioning principles.

CLARANS [11] relies on a randomized search of a graph to find medoids representing clusters. The algorithm takes as input maxneighbor and numlocal, selects a random node and then checks a sample of the neighbors of the node. If a better neighbor is found based on the “cost differential of the two nodes” it moves to the neighbor and continues processing until the maxneighbor criterion is met. Otherwise, it declares the current node a local minimum and starts a new pass to search for other local minima. After a specified number of local minima (numlocal) are collected, the algorithm returns the best of these local values as the medoid of the cluster.

<table>
<thead>
<tr>
<th>Proximity</th>
<th>Forms</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min-</td>
<td>$D_{mk} = \sum_{i=1}^{d}</td>
<td>X_i - Y_i</td>
</tr>
<tr>
<td>Euclidean</td>
<td>$D_{eu} = \sum_{i=1}^{d}</td>
<td>X_i - Y_i</td>
</tr>
<tr>
<td>City-block</td>
<td>$D_c = \sum_{i=1}^{d}</td>
<td>X_i - Y_i</td>
</tr>
<tr>
<td>Sup distance</td>
<td>$D_{sp} = \max</td>
<td>X_i - Y_i</td>
</tr>
<tr>
<td>Mahalanobis distance</td>
<td>$D_m = (X - Y)^T S^{-1} (X - Y)$, where $S$ is within group co-variance matrix</td>
<td>Invariant to any non-singular linear transformation. $S$ is calculated based on all objects. Tends to form hyper-ellipsoidal clusters. When features are not correlated, squared Mahalanobis distance is equivalent to squared Euclidean distance. May tend to be computationally expensive.</td>
</tr>
<tr>
<td>Pearson correlation</td>
<td>$D_p(X,Y) = \sum_{i=1}^{d} \frac{(X_i - Y_i)^2}{Y_i}$</td>
<td>Not a metric. Derived from correlation coefficient. Unable to detect the magnitude of differences of two variables.</td>
</tr>
</tbody>
</table>

The effective use of any of these proximity measures (Table 1) is highly influenced by the number of data records/instances, their dimensionality, the level of precision required and the nature of analysis required.

IV. GENE-BASED CLUSTERING APPROACHES

In this section, we review a series of clustering algorithms based on their expression patterns. The purpose of gene-based clustering is to group together co-expressed genes which indicate co-function and co-regulation. We will first review a series of clustering algorithms belonging to various approaches as reported in Section I-C, which have been applied to group genes. Finally, we present the issues involved in gene-based clustering.
Discussion: Though partitioning based clustering algorithms can find separate clusters, in the context of gene expression data mining, it has the following disadvantages.

i. The number of clusters is not known apriori.
ii. The proximity measures used by most of these methods are inadequate due to high dimensionality of gene data.
iii. The gene data, apart from displaying disjoint patterns of clusters, often show evidence of intersected and embedded cluster patterns, which are usually not detectable by the partitioning approach.

B. Hierarchical Approaches

Hierarchical clustering generates a hierarchical series of nested clusters which can be graphically represented by a tree, called dendrogram. We can obtain a specified number of clusters by cutting the dendrogram at an appropriate level.

Hierarchical clustering algorithms can be further divided into agglomerative approaches and divisive approaches based on how the dendrogram is formed. Agglomerative algorithms (bottom-up approach) perform repeated amalgamations of groups of data until some pre-defined threshold is reached, whereas divisive algorithms (top-down approach) recursively divide the data until some pre-defined threshold is reached. For agglomerative approaches, linkage is used as the criteria to determine the distance between two clusters. Single linkage is the smallest minimum distance between two objects of two clusters, whereas complete linkage is the smallest maximum distance between two objects of the two clusters and average linkage is the mean distance between every pair of objects of two clusters.

CURE [12] is an agglomerative hierarchical clustering algorithm, which begins by choosing a constant number c, of well scattered points, from a cluster. These points are used to identify the shape and size of the cluster. The next step of the algorithm shrinks the selected points toward the centroid of the cluster using some predetermined fraction. ROCK [13], also an agglomerative hierarchical clustering algorithm, employs links and not distances, to measure the similarity/proximity between a pair of data points, before merging them. CHAMELEON [14] uses a graph partitioning algorithm to partition the data using a method based on the k-nearest neighbor approach and then uses an agglomerative hierarchical clustering algorithm to combine the sub-clusters and find the real clusters. A study of the CHAMELEON algorithm using gene data is given in [83].

UPGMA [15] adopts an agglomerative method to graphically represent the clustered dataset. In this method, each cell of the gene expression matrix is colored on the basis of the measured fluorescence ratio and the rows of the matrix are reordered based on the hierarchical dendrogram structure and a consistent node-ordering rule. After clustering, the original gene expression matrix is represented by a colored table (a cluster image) where large contiguous patches of color represent groups of genes that share similar expression patterns over multiple conditions. However, it is not robust in the presence of noise.

BIRCH [16] is an integrated hierarchical clustering method that performs cluster representations by introducing two new concepts, clustering feature (CF) and clustering feature tree, which help the clustering method to: (i) achieve good speed (ii) scalability in large databases and, (iii) better utilization of memory. BIRCH is also effective for incremental and dynamic clustering of incoming objects. It applies a multiphase clustering technique: a single scan of the dataset yields a basic good clustering, and additional scan(s) may be used to further improve the quality of clustering.

AMOEBA [17] is an algorithm that supports hierarchical spatial clustering based on Delaunay triangles [17].

Discussion: Hierarchical clustering methods are vastly popular because of their presentation of cluster results, which biologists often prefer, as is evidenced by [50], [56]-[58], [61]-[68]. However, the effectiveness of a method in this approach is mostly influenced by the following points.

i. The appropriateness of the proximity measure used.
ii. The scope for incorporating regulation information, expression values and other type of biological knowledge while expanding the cluster.
iii. Ability to work with high dimensional numeric data.
iv. Representation of nested cluster structure is clear but inadequate in representing intersected clusters.

C. Density Based Approaches

Density based clustering identifies dense areas in the object space, where clusters are highly dense areas separated by sparsely dense areas. The given cluster continues to grow as long as the density (number of objects or data points in the “Eps-neighborhood”) exceeds some threshold. Such a method can be used to filter out noise and discover clusters of arbitrary shape. Moreover, it has been proposed in [18] that statistically significant patterns can be derived from dense regions, which can then be used to identify genes and/or samples of interest and also eliminate genes and/or samples corresponding to outliers, noise, or abnormalities.

The well known DBSCAN [19] density-based clustering algorithm starts with an arbitrary instance p in a dataset D and retrieves all instances of D with respect to Eps and MinPts. The algorithm makes use of a spatial data structure R*-tree to locate points within Eps distance from the core points of the clusters. It is found to be effective over spatial data with uniform density. Its use in gene data clustering is given in [82]. An extension to DBSCAN is OPTICS [20] that relaxes the strict requirements of input parameters and can handle variable density. OPTICS computes an augmented cluster ordering to automatically and interactively cluster the data. The ordering represents the density based clustering structure of the data and contains information that is equivalent to density-based clustering obtained by a range of parameter settings. DBCLASD [21] is another locality-based clustering algorithm, but unlike DBSCAN, the algorithm assumes that the points inside each cluster are uniformly distributed. Each cluster has a probability distribution of points to their nearest neighbors, and this probability set is used to define the cluster. A grid-based representation is used to approximate the
clusters as part of the probability calculation. The major advantages of DBCLASD are that it requires no external input, it is unaffected by noise since it relies on a probability based distance factor and it can handle clusters of arbitrary shape. Its main disadvantages are that it assumes uniformly distributed points in a cluster, limiting the effectiveness of the algorithm. Its runtime slows as the size of the database increases and it is computationally expensive when compared to later algorithms.

DENCLUE [22] is a generalization of partitioning, hierarchical, density-based and grid-based clustering approaches. The algorithm initially performs a pre-clustering step creating a map of the active portion of the dataset which is used to speed up density function calculations. The next step is clustering, including the identification of density-attractors and their corresponding points. DENCLUE can identify clusters of arbitrary shape and exhibits good clustering properties in presence of noise.

TURN* [23] consists of an overall algorithm and two component algorithms: an efficient resolution dependent clustering algorithm, TURN-RES, which returns both a clustering result and cluster features from that result, and TURN-CUT, an automatic method for finding the important or “optimum” resolutions from a set of resolution results from TURN-RES. TURN* removes the need for input parameters.

The basic idea of the DHC [24] method is to consider a cluster as a high-dimensional dense area, where data objects are “attracted” to each other. It then organizes the cluster structure of the dataset in a two-level hierarchical structure. Initially, the root node represents the entire data as a single dense area of the density tree which is then split into several subdense areas based on some criteria, represented by a child node of the root node. These subdense areas are further split, until each subdense area contains a single cluster. The computational complexity of this step makes DHC inefficient.

**Discussion:** Most existing density based algorithms are capable of handling two-dimensional data with uniform global density distribution. However, the characteristics of gene expression data demand a density based clustering algorithm with the following capabilities:

i. The algorithm should be able to handle high dimensional numeric data.
ii. It should recognize embedded and intersected cluster patterns apart from recognizing uniformly distributed cluster patterns.
iii. It should be less dependent on input parameters.

**D. Model Based Approach**

Based on our study and analysis of gene expression data [72]-[76], we observe that often a single gene may exhibit membership in more than one cluster. Hence it demands a probabilistic clustering approach for a better fit. Model based clustering is one such approach which is suitable for a problem like gene expression data. Such an algorithm attempts to optimize the fit between the given data and some mathematical model. Applications of model based approach to gene expression data are discussed in [56], [57], [69]-[71].

COBWEB [25] creates hierarchical clustering in the form of a classification tree, where the input objects are described by categorical attribute-value pairs. It also has two operators: merging and splitting, which help COBWEB to become less input order dependent and also allow it to perform bi-directional search.

The Self-Organizing Map (SOM) [26] was developed on the basis of a single layered neural network. 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. In the process of running the algorithm, each data object acts as a training sample which directs the movement of the reference vectors towards the denser areas of the input vector space, so that those reference vectors are trained to fit the distributions of the input dataset. When the training is complete, clusters are identified by mapping all data points to the output neurons. The advantage of SOM is that (i) it generates intuitive cluster patterns of a high-dimensional dataset and (ii) it is non-susceptible to noisy data. Some disadvantages of SOM are that (i) the number of clusters and the grid structure of the neuron map need to be given as input, and (ii) it is sensitive to the input parameter. SOM is a popular method for clustering data in biological context and can be found in [54], [55], [59], [60], [61], [63] and [77].

AutoClass [27] uses the Bayesian approach, starting from a random initialization of the parameters, incrementally adjusting them in an attempt to find their maximum likelihood estimates. It also assumes that, in addition to the observed or predictive attributes, there is a hidden variable which reflects the cluster membership for every element in the dataset. Therefore, the data-clustering problem is also an example of supervised learning from incomplete data due to the existence of such a hidden variable. The use of AutoClass for clustering microarray data is shown in [58] and [78].

**Discussion:** Though the model based approach can be considered more relevant to the gene expression data mining problem, most existing methods under the approach suffer from the following disadvantages:

i. the number of clusters and the grid structure need to be given as input,
ii. they are sensitive to the input parameter, and
iii. the algorithms are not cost effective.

**E. Graph Theoretic Approach**

Graph theoretical clustering techniques explicitly work with data represented in terms of a graph. AUTOCLUST [28] automatically extracts boundaries based on Voronoi modeling and Delaunay Diagrams. Parameters need not be specified by users but are revealed from the proximity structures of the Voronoi modeling, and AUTOCLUST calculates them from the Delaunay Diagram. This removes human-generated bias and also reduces the exploration time. The advantages are (i) It is effective in the detection of clusters of different densities, and (ii) It identifies and removes multiple bridges linking clusters.
CLICK [29] tries to identify clusters as a highly connected component in a proximity graph based on a probabilistic assumption. It defines the weight of an edge as the probability that the vertices of the edge are in the same cluster and then iteratively finds the minimum cut in the graph and recursively splits the dataset into a set of connected components subject to a predefined threshold. It also includes two post-pruning steps to refine cluster results. CLICK is capable of recognising intersecting clusters. It also generates better quality clusters over gene expression datasets in terms of homogeneity and separation. It is found in literature in [43], [59], [69] and [79].

CAST [30] is based on the concept of a corrupt clique graph data model. It takes as input a real, symmetric, n-by-n similarity matrix $M$ and an affinity threshold $t$. It assumes that the input dataset contains errors and that the true clusters of the data points are obtained by a disjoint union of complete sub-graphs where each clique represents a cluster. A cluster is defined as a set of high affinity elements subject to $t$, referred to as a clique graph. A proximity graph is derived (based on an n-by-n similarity matrix) using the clique graph by flipping each edge/non-edge with some probability measure. Thus, the clustering process defined by CAST is a process of elimination of noise or contamination from the corrupt version with minimum number of flips. CAST discovers clusters one at a time. It is popular in use of generating clusters in gene expression data [30], [43], [50], [59], [62], [69] and [80].

Discussion: The graph theoretic approach can be considered more relevant to gene expression data mining as it is capable of discovering intersected and embedded clusters. However, it sometimes generates non-realistic cluster patterns. Moreover the results are dependent on proximity measures, so inappropriate selection of the proximity measure will result in non-biologically relevant clusters.

F. Soft Computing Approach

Fuzzy Clustering: Gene expression data analysis sometimes encounters an intersecting gene pattern in which case a crisp or hard clustering may not yield a good result. However for fuzzy clustering, a gene can belong to several clusters with certain degrees of membership. An application of fuzzy clustering in biological context can be found in [81].

Fuzzy c-Mean (FCM), which is a generalisation of ISODATA [31], operates over numeric data to identify c-fuzzy clusters (c is user input). FCM uses Euclidean distance and encounters problems in outlier handling, identification of initial partitions and discovery of clusters of all shapes, which are also the limitations of hard-partitioning based clustering techniques. However several variants of FCM are found in the literature that attempt to overcome these limitations.

Neural Networks-Based Clustering: Two popular Neural Networks-based clustering approaches, SOFM (Self Organizing Feature Map) and ART (Adaptive Resonance Theory), are discussed briefly here.

SOFM can be viewed as a visual representation of a lattice data structure of neurons, interconnected via adaptable weights. During the training process, the neighbouring input patterns are projected into the lattice corresponding to adjacent neurons. The advantages of SOFM are: (i) It enjoys the benefits of input space density approximation and, (ii) it is input order independent. The disadvantages are (i) like k-means, SOFM needs to predefine the size of the lattice, (the number of clusters) and, (ii) it may suffer from input space density misrepresentation. To overcome the limitations of SOFM, several variants of SOFM can be found in the literature [34] - [36]. The use of SOFM for clustering biological data is found in [82].

ART, a large family of NN architectures, is able to learn any input pattern in a fast, stable and self-organising way. ART1 deals with binary input patterns and can be extended to arbitrary input patterns by using a coding mechanism, whereas ART2 extends the application to analog input patterns, and ART3 achieves efficient parallel search in hierarchical structures based on certain biological processes. Several variants of the use of ART can be found in literature [61].

GA Based Clustering: GA (Genetic Algorithm) clustering is randomized search and optimization techniques based on the principles of evolution and natural genetics. GAs perform search in complex, large and multimodal landscapes, and provide near optimal solutions for the objective function of an optimization problem. Several GA-based clustering algorithms are found in the literature [37] – [42].

GenClust [43] is an effective GA-based clustering algorithm with two key features (i) a simplified and compact coding of the search space and, (ii) is used in conjunction with a data driven internal data validation method. GenClust works stage-wise and produces a sequence of partitions consisting of $k$ classes. The process of partitioning into classes continues until a termination criterion is satisfied. The GenClust algorithm is used in [59] to cluster gene expression data.

Discussion: In case of fuzzy clustering, the problem of specifying the number of clusters remains. The limitations of FCM can be overcome by (i) by using City Block Distance ($L_1$ norm) to improve the robustness of FCM to outliers [32] (ii) Initialization strategy of unsupervised tracking of cluster prototypes in a 2-layer clustering scheme [33] (iii) use of appropriate prototypes and distance functions for identification of clusters of all shapes. The basic advantage of ART is that it is fast, exhibits stable learning and pattern detection. The disadvantage is its (i) inefficiency in dealing with noise, and (ii) deficiency in higher dimension representation for clusters. GenClust creates good clusters from gene expression datasets by using external (when a true solution is available) and internal (when a true solution is not known) criteria. However, like the other GA-based solutions, GenClust is also not free from the local optima problem.

G. Incremental Approach

In [44], the authors present an incremental clustering algorithm based on the DBSCAN algorithm. A one pass clustering algorithm for relational datasets is proposed in [45]. Rough set theory is employed in the incremental approach for clustering interval datasets in [46]. In [47], an incremental genetic k-means algorithm is presented. In [48], an incremental gene selection algorithm using a wrapper-based
method that reduces the search space complexity since it works on the ranking directly, is presented. In [49], an incremental clustering algorithm over gene expression data is presented; it uses regulation information to store the cluster information for use when clustering genes incrementally.

The effectiveness of an incremental gene clustering algorithm depends upon the proximity measure used and the differing density.

V. CLUSTER VALIDITY MEASURES

The previous section has reviewed a number of clustering algorithms that partition the dataset based on a variety of clustering criteria. For gene expression data, clustering results in groups of co-expressed genes, groups of samples with a common phenotype, or “blocks” of genes and samples involved in specific biological processes. However, different clustering algorithms, or even a single clustering algorithm using different parameters, generally result in different sets of clusters [50]. Therefore, it is important to compare various clustering results and select the one that best fits the “true” data distribution. Cluster validation is the process of assessing the quality and reliability of the cluster sets derived from various clustering processes.

Generally, cluster validity has three aspects. First, the quality of clusters can be measured in terms of homogeneity and separation on the basis of the definition of a cluster. Objects within one cluster are similar to each other and different from objects in other clusters. The second aspect comes from “ground truth” of the clusters. The “ground truth” could come from domain knowledge, such as the clinical diagnosis of normal or cancerous tissues. Cluster validation is based on the agreement between clustering results and the “ground truth”. The third aspect focuses on the reliability of the clusters or the likelihood that the cluster structure is not formed by chance. In this section, we will discuss these three aspects of cluster validation in the light of some popular validity measures.

A. Rand Index

Rand index [84] is a measure of the similarity between two clusters. The Rand index is defined as the number of pairs of objects that are either in the same group or in different groups in both partitions divided by the total number of pairs of objects.

\[
\text{Rand index} = \frac{a + d}{a + b + c + d}
\]

Where \(a\) is the number of object pairs \((g_i, g_j)\), where \(C_g = 1\) and \(P_g = 1\), \(b\) is the number of object pairs \((g_i, g_j)\), where \(C_g = 1\) and \(P_g = 0\), \(c\) is the number of object pairs \((g_i, g_j)\), where \(C_g = 0\) and \(P_g = 1\), \(d\) is the number of object pairs \((g_i, g_j)\), where \(C_g = 0\) and \(P_g = 0\).

The Rand index lies between 0 and 1. The maximum value i.e., 1 is achieved when both partitions, C and P, agree perfectly. Some examples of papers that use Rand index can be found in [57], [63] and [70].

B. Cluster Homogeneity

Homogeneity measures the quality of clusters on the basis of the definition of a cluster: objects within a cluster are similar while objects in different clusters are dissimilar. Homogeneity measure used in this section is that of the overall average homogeneity used in [85]. It is calculated as follows.

i. Compute the average value of similarity between each gene \(g\) and the centroid of the cluster \(C\), to which it has been assigned.

\[
H(C) = \frac{1}{|C|} \sum_{g_i \in C} \text{Similarity}(g_i, g)
\]

where \(g_i\) is the centroid of \(C\).

ii. Calculate the average homogeneity for the clustering \(C\) weighted according to the size of the clusters as

\[
H_{\text{avg}} = \frac{1}{|G|} \sum_{C \in C} |C| H(C)
\]

C. Silhouette Index:

Silhouette index [86] is used to assess the quality of any clustering solution. This index reflects the compactness and separation of clusters. It is calculated as follows.

i. Compute \(a(g)\), i.e., the average distance of gene \(i\) to the the other genes of cluster \(A\) to which it belongs.

ii. Compute \(d(g_i, C_j)\) where \(d(g_i, C_j)\) is the average distance of gene \(g\), from the genes of cluster \(C_j\) where \(g \in C_j\).

iii. Compute \(b(g)\), where \(b(g) = \min\{d(g_i, C_j)\}\) where \(C = \{C_1, C_2, ..., C_n\}\) and \(A \in C\), i.e., \(b(g)\) represents the distance of gene \(g\) to its closest cluster. Now compute the silhouette width of gene \(g\) as

\[
S(g) = \frac{b(g) - a(g)}{\max\{a(g), b(g), 0\}}
\]

iv. Compute silhouette index by finding the average of \(S(i)\) over \(i = 1, 2, ..., G\), where \(G\) is the total number of genes: \(S = \text{average}\{S(g)\}\).

The value of silhouette index varies from -1 to 1 with higher values indicating better clustering. Silhouette index has been used in [63].

D. Z-score:

Z-score [87] is calculated by investigating the relation (Mutual Information - MI) between a clustering obtained by an algorithm and the functional annotation of the genes in the cluster. A higher value of z-score indicates that genes are better clustered by function, indicating a more biologically relevant clustering result. The z-score represents a standardized distance between the MI value obtained by clustering and those MI values obtained by random assignment of genes to clusters. Higher z-scores indicate that the clustering results are more significantly related to the gene function. The use of z-score is shown in [88].

E. Jaccard Coefficient:

The Jaccard Coefficient measures the proportion of pairs that are in the same cluster C and in the same partition P from those that are either in the same cluster or in the same partition [91]. The Jaccard Coefficient is defined as

\[
J = \frac{a}{a + b + c}
\]

where \(a\) = SS if the pair belongs to the same cluster C and to the same group P, \(b\) = SD if the pair belongs to the same
cluster C and to different groups P and e = DS if the pair belongs to different clusters C and to the same group P.

As in the Rand Index, the values of these coefficients lies between 0 and 1, and values close to 1 indicate height agreement between C and P.

F. Davies-Bouldin Index:

Let s_i be a measure of dispersion of cluster C_i and d(C_i,C_j) = d_{ij} the dissimilarity between two clusters. The dispersion of a cluster C_i is defined as s_i = √(|x - x'|^2).

The Davies-Bouldin index [92] is defined as DB_m = \frac{1}{m} \sum_{i=1}^{m} R_i, with \, R_i = max_{j=1,...,m}d_{ij}, \, i = 1, ..., m and \, R_0 is a similarity index between C_i and C_j satisfying the condition R_{ij} = \frac{s_i + s_j}{d_{ij}}.

G. Dunn Index:

The Dunn Index [93] is defined as:

D_m = \min_{i=1,...,m} \left\{ \frac{d(C_i,C_j)}{\max_{k=1,...,m}diam(C_k)} \right\}

H. P-value:

The reliability of the resulting clusters can be estimated by the p-value [52] of a cluster. It measures the probability of finding the number of genes involved in a given Gene Ontology (GO) term (i.e., function, process and component) within a cluster. From a given GO category, the probability p of getting k or more genes within a cluster of size n, is defined as:

P = 1 - \sum_{i=0}^{k-1} \binom{n}{i} \frac{1}{\binom{p}{i}}

A cumulative hyper-geometric distribution is used to compute the p-value. A low p-value indicates that the genes belonging to the enriched functional categories are biologically significant in the corresponding clusters. The use of p-value to compare different clustering solutions of yeast cell-cycle gene expression data is given in [90].

VI. DATASETS USED

We present some datasets related to Human, Yeast and Rat Central Nervous System in Table 2 along with their sources. These datasets have often been used to evaluate the clustering of gene array data.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>No. of genes</th>
<th>No of conditions</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yeast Sporulation</td>
<td>6118</td>
<td>7</td>
<td><a href="http://cmgm.stanford.edu/pbrown/sporulation">http://cmgm.stanford.edu/pbrown/sporulation</a></td>
</tr>
<tr>
<td>Yeast Cell Cycle</td>
<td>698</td>
<td>72</td>
<td>Sample input files in Expander [115]</td>
</tr>
<tr>
<td>Subset of Yeast Cell Cycle</td>
<td>384</td>
<td>17</td>
<td><a href="http://faculty.washington.edu/kayee/cluster">http://faculty.washington.edu/kayee/cluster</a></td>
</tr>
<tr>
<td>Rat CNS</td>
<td>112</td>
<td>9</td>
<td><a href="http://faculty.washington.edu/kayee/cluster">http://faculty.washington.edu/kayee/cluster</a></td>
</tr>
<tr>
<td>Subset of Human Fibroblasts</td>
<td>517</td>
<td>13</td>
<td><a href="http://www.sciencemag.org/feature/data/984559.hsl">http://www.sciencemag.org/feature/data/984559.hsl</a></td>
</tr>
<tr>
<td>Human cancer data</td>
<td>9703</td>
<td>60</td>
<td><a href="http://discover.nci.nih.gov/datasets.jsp">http://discover.nci.nih.gov/datasets.jsp</a></td>
</tr>
<tr>
<td>Colon cancer data</td>
<td>6500</td>
<td>40 and 22</td>
<td><a href="http://datam.i2r.astar.edu.sg/datasets/krbd/ColonTumor/ColonTumor.html">http://datam.i2r.astar.edu.sg/datasets/krbd/ColonTumor/ColonTumor.html</a></td>
</tr>
</tbody>
</table>

VII. CHALLENGES IN CLUSTERING GENE EXPRESSION DATA

Clustering gene expression data poses challenges that are different from those of clustering non-biological data. This is due to the very nature of data being collected from microarray experiments.

A. Challenges and Research Issues

Studies have confirmed that clustering algorithms are useful in identifying groups of co-expressed genes and discovering coherent expression patterns. However, due to the distinct characteristics of time-series gene expression data and the special requirements from the biology domain, clustering gene expression data still faces the following challenges.

1) The effectiveness of a clustering technique is highly dependent on the proximity measure, used by the technique. Choosing or finding an appropriate proximity measure is a challenging task.

2) Most existing clustering techniques are either dependent on input parameter(s) or stopping criteria for discovery of the “true” number of clusters, which is a major task and hence a challenge.

3) Gene expression data often contain clusters which are “highly connected” [89], intersected or even embedded [24]. Hence the clustering algorithm should be capable of effectively managing such a situation.

4) The available gene datasets often contain a lot of noise and missing values. Thus a clustering algorithm should be capable of extracting the “true” number of clusters in the presence of this noise and also be able to handle missing values.

5) A clustering algorithm should be efficient in order to scale with the increasing size of datasets as well as dimensionality.
6) Apart from clustering, an algorithm should be capable of showing the associations among the clusters which may be useful for drawing conclusions.

In the case of partitioning approaches, algorithms like PAM or Fuzzy C-Means have been observed to be robust. However, they suffer from limitations such as (i) the number of clusters are to be known *apriori*, (ii) the proximity measure used may be inadequate in finding the ‘true’ number of biologically relevant clusters. Similarly, the algorithms following the hierarchical approach can be found advantageous from the biologists’ perspectives as they help to represent the cluster-cluster association, apart from individual co-expressed gene group representation. However, they also suffer from limitations such as (i) difficulty in deciding the appropriate stopping criteria, and (ii) simultaneous representation of disjoint, embedded and intersected clusters.

Density approaches have already been established as being good at finding clusters of all shapes. They are capable of identifying global as well as local (embedded) clusters. However, two limitations of such approaches are: (i) They are input parameter sensitive, and (ii) ineffective in finding intersected patterns over high dimensional data. A model-based approach provides an estimated probability that a single gene may exhibit membership in more than one cluster indicating that a gene may have a high correlation with two totally different clusters. It discovers good values for its parameters iteratively and can handle various shapes of data. However, it suffers from the limitations that (i) it can be computationally expensive since a large number of iterations may be required to find its parameters, and (ii) it assumes that the dataset fits a specific distribution which is not always true.

Graph theoretic algorithms are suitable for subspace and high dimensional data clustering. The algorithms under this approach do not require a user-defined number of clusters, can handle outliers efficiently and they are capable of discovering intersected and embedded clusters. However, they are limited by (i) the difficulty faced in determining a good threshold value, and (ii) they require *apriori* knowledge of the dataset. Among the soft computing approaches, Fuzzy C-Means (FCM) and Genetic Algorithms (GA) have been used effectively in clustering gene expression data. The Fuzzy C-Means algorithm requires the number of clusters as an input parameter. The GA based algorithms can detect biologically relevant clusters but are dependent on proper tuning of the input parameters.

VIII. Conclusion

There are two groups of people who are involved in the clustering of biological data. One is the biologist who uses an existing clustering algorithm to solve an underlying biological problem. The challenge before the biologist is to make an appropriate choice of an algorithm since different algorithms will produce different results. The other is the developer of clustering algorithms, who consistently strives to improve existing algorithms, so that the underlying biological problems can be solved efficiently. A proper amalgamation of these two groups will lead to rapid advancement in this field.

Although gene expression clustering has been done by applying *k*-means, hierarchical clustering and SOMs algorithms, the desired features for clustering include minimum user input, finding arbitrary shaped clusters, robustness to outliers and the ability to handle higher dimensionality. If outliers pose a problem while using *k*-means for clustering gene expression data, the biologist can consider an improvement by opting for *k*-medoids, CLARAnS, CLARANS as alternatives. If hierarchical clustering is to be used, then BIRCH is suitable for outlier detection while CURE or DHC could be used to detect arbitrary shaped clusters. Fuzzy C-means or GA based approach supports overlapping clusters of co-regulated genes, while projected density-based approaches improve quality on high-dimensional data but again needs user specified parameters.

A clustering algorithm’s suitability to cluster biological data depends upon certain desirable features such as speed, minimum number of input parameters, robustness to noise and outliers, redundancy handling and independence of object order input. Though the features of many clustering algorithms match these requirements, they have not yet been applied to clustering biological data. Moreover, not all validity measures are suitable for all gene datasets; hence a judicious choice of the applicability of the validity measure has to be made.

It is well known that most clustering methods are highly sensitive to input data and a slight variation or change in the data may result in very different gene clusters. If the information from genomic knowledge bases, such as GO, could be incorporated (data fusion) earlier in the analysis of genomic data, that additional information about genes and their relationship with each other will improve stability, accuracy and/or biological relevance of the cluster results.

In this survey, an attempt has been made to provide a comprehensive survey of various clustering approaches in the context of coherent pattern identification in gene expression data. Effectiveness of a clustering technique is highly influenced by the proximity measure used by the technique. This survey also presents a partial list of proximity measures available for numeric data clustering. It attempts to provide analysis of the effectiveness of algorithms belonging to a specific approach in gene expression data mining. Finally, it discusses the various validity measures and the sources of various datasets for the effective evaluation of clustering results.

REFERENCES


[82] C. Bohm, K. Kaling, H.-P. Kriegel and P. Kroger, “Density Connected Clustering with Local Subspace Preferences”, In Proc. 4th IEEE Int. Conf. on Data Mining (ICDM’04), Brighton, UK, 2004


