Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB 2007)

Discovering Connected Patterns in Gene Expression Arrays

Noha A. Yousri[*†], Mohamed A. Ismail[*], and Mohamed S. Kamel[†], *Fellow, IEEE*
[*]Computers and System Engineering, University of Alexandria, Alexandria, Egypt
[†]Electrical and Computer Engineering, University of Waterloo, Ontario, Canada
{nyousri,mkamel@pami.uwaterloo.ca}

***Abstract*- Clustering methods have been extensively used for gene expression data analysis to detect groups of related genes. The clusters provide useful information to analyze gene function, gene regulation and cellular patterns. Most existing clustering algorithms, though, discover only coherent gene expression patterns, and do not handle connected patterns. Coherent and connected patterns correspond to globular and arbitrary shaped clusters, respectively, in low dimensional spaces. For high dimensional gene expression data, two connected patterns can be two similar patterns with time lags in a time series data, or in general, two different patterns that are connected by an intermediate pattern that is related to both of them. Discovering such connected patterns has important biological implications not revealed by groups of coherent patterns. In this paper, a novel algorithm that finds connected patterns, in gene expression data, is proposed. Using a novel merge criterion, it can distinguish clusters based on distances between patterns, thus avoiding the effect of noise and outliers. Moreover, the algorithm uses a metric based on Pearson correlation to find neighbours, which renders it a lower complexity than related algorithms. Both time series and non temporal gene expression data sets are used to illustrate the efficiency of the proposed algorithm. Results on the serum and the leukaemia data sets reveal interesting biologically significant information.**

I. INTRODUCTION

Gene expression cluster analysis includes a wide range of algorithms, either specially designed for this application, or other algorithms already used in pattern recognition and data mining. The most common clustering techniques used for gene expression analysis are: k-Means (see [16], [2]), Kohonen SOM (Self Organizing Maps) used in [4], hierarchical clustering used in [12], Consensus Clustering [3], and recent developed algorithms as CAST [1], HCS [19], and CLICK [2]. Each of these clustering algorithms has its objective, thus each leads to a somehow different clustering result. Other techniques are also used for exploring gene expression sets as SPIN [7] and the interactive tool in [17].

Clustering algorithms such as k-Means, SOM, average linkage hierarchical clustering, CLICK and CAST are different approaches for discovering coherent groups of patterns. The authors of CLICK [2], for instance, use a homogeneity-separation validity measure to measure the quality of the clustering solution which is only suitable for coherent groups of patterns. On the other hand discovering a group of connected patterns in gene expression data has not been given much attention.

Coherent patterns have their own biological implications. Genes that are up/down regulated (co-expressed) together might be functionally related. However, as discussed above, many algorithms, even those not related to gene expression analysis can obtain this type of co-regulation between gene expressions. Obtaining coherent expression patterns in high dimensional gene expression corresponds to obtaining clusters of globular or hyperspherical shapes.

This research tackles the problem of obtaining connected gene expression patterns, rather than coherent patterns. Pattern connectivity corresponds to the known arbitrary-shaped clustering in low dimensions, and has different biological implications. Most important is discovering the impact of one gene's expression on another gene's expression in time series data. There can be time lags between one gene's expression and a related gene's reaction to it, which cannot be discovered by clusters of coherent patterns. In non temporal patterns, connected patterns can reveal genes that are related to a number of coherent groups at the same time, revealing new biological aspects.

SPIN (Sorting Points Into Neighbourhoods) [7], as well as dimensionality reduction techniques including SVD (Singular Value Decomposition) used in [8], were used to detect connected patterns in gene expression data sets. SPIN is an expensive technique of a cubic term in the data size. Moreover, it is not a clustering method, and does not put any measures for clustering. Hence, it cannot avoid the effect of outliers in data, and cannot differentiate low and high dense regions in the data.

Hotler et. al [8], and Alter et. al [20] used SVD to discover continuity in patterns of gene expression. Yet, those techniques are computationally expensive with complexity of at least a quadratic term in the data size. They are only used for exploring the structure of the data, rather than producing clusters.

Other related algorithms that find arbitrary shaped clusters in low dimensions are single linkage, DBScan [13], DenClue [24], WaveCluster [22] and Chameleon [14]. Yet, only Chameleon can find arbitrary shaped clusters of different densities in the same data set. However, it has been applied for low dimensions, and suffers some drawbacks as its slow speed and difficulty in determining the parameters.

The algorithm proposed here "Mitosis" has been studied and compared (refer to [18] for results on 2-D datasets) to other clustering algorithms that were applicable in low dimensions such as DBScan [13] and Chameleon [14]. The algorithm is

much faster than Chameleon, and more efficient than DBScan in discovering clusters of different densities (see DBScan results for DS5 dataset in [18], where it fails to discover all clusters) The algorithm uses a dynamic model of clustering that is able to detect the structure of the data. Mitosis is able to discover clusters of arbitrary shape and arbitrary density, which corresponds to finding clusters of connected expression patterns with variable cluster densities. Its ability to measure internal distance structure enables it to separate lower dense areas from higher dense ones, avoiding the effect of noise and outliers. Moreover, it is able to maintain a relatively low time complexity.

II. COHERENT PATTERNS VS. CONNECTED PATTERNS

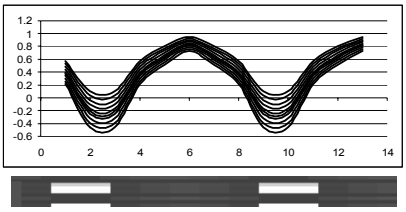


Figure 1: Coherent patterns, and the corresponding color gradient.

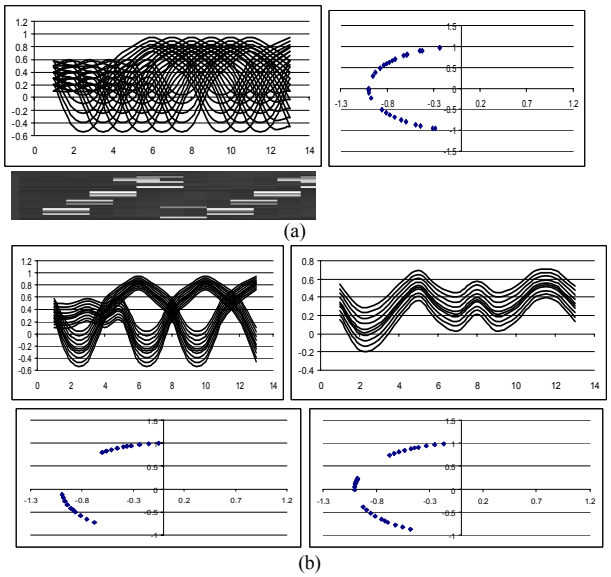


Figure 2: Illustration of connected patterns (a) Similar expression patterns with time lags shown in a connected half circle of the SVD two modes, and the diagonal transition in the color gradient, (b) Illustration of nontemporal connectedness.

The difference between coherent patterns obtained by the majority of clustering algorithms, and connected patterns discussed here, is illustrated in figures 1 and 2. The first example in fig. 1, plots a group of coherent expression patterns (upper graph), and the corresponding colour-gradient representation (rows of the data matrix where brightest black/white colour corresponds to highest/lowest expression values). Fig. 2-a plots a group of connected expression patterns, followed by the corresponding normalized coefficients of the two highest ranked SVD modes, and the colour-gradient representation. Their colour-gradient scheme shows a transitional behaviour in gene expression patterns. This is reflected by the continuity in relation (half circle) between the SVD modes' coefficients. This is a case of a group of similar expression patterns with time lags, assuming gene expression time series. Whereas, a group of connected patterns in non-temporal gene expression arrays can be as the example shown in fig. 2-b, where two groups of coherent patterns, shown in the left graph, are related by other intermediate patterns, shown in the right graph of the same figure. The SVD representation, in the lower plots of fig. 2-b, shows that the two groups of coherent patterns (left plot) have a gap between them. While, when introducing the intermediate patterns, the gap starts to diminish, and a connection between the two discrete groups starts to develop (right plot).

The authors of SPIN [7], and Hotler et. al [8] point out the importance of discovering continuous gene expression patterns. In SPIN, this is achieved by permuting the similarity matrix until obtaining a matrix arrangement that uncovers the structure of the data. The matrix arrangement of a connected shape (e.g. circle or elongated shape) in a 2 dimensional space is compared to the permuted similarity matrix of gene expression patterns in high dimensions. The comparison showed that the high dimensional gene expression patterns were connected in a manner similar to the 2-D shapes. The technique was applied to Spellman's yeast cell cycle data set, revealing a circular shape of connected patterns for 500 genes. Also, genes from Leukaemia's ALL B lineage and ALL T lineage revealed connectivity in a shape similar to an elongated bar.

In [8], SVD is used to analyse gene expression time series. Connected patterns were revealed when plotting the coefficients of the two highest ranked modes against each other. The authors used yeast cell cycle and serum data sets, and pointed out the continuity of pattern expression from the SVD modes. This continuity in expression patterns is important for revealing genetic pathways, that can be used to build gene networks from expression data [5].

Alter et. al [20] used SVD to reveal the continuity of expression patterns for cell cycle data. The colour-gradient representation shown in that work, clarifies the presence of connectivity by showing a transition in gene expression patterns along the time points.

III. PROPOSED ALGORITHM

"Mitosis" is proposed here for finding arbitrary shaped clusters of arbitrary densities in gene expression patterns. The proposed distance measures together with the proposed distance-based clustering criteria are able to efficiently discover connected

patterns. This section presents the clustering measures and criteria used to find the clusters.

A. Nearest Neighbours

Similar to DBScan and Chameleon, neighbourhood information is used to reveal connectivity between patterns. To fetch nearest neighbours in high dimensions, a suitable distance metric is used together with a metric tree as that proposed in [25].

Pearson Correlation based Metric: This is a proposed metric for finding nearest neighbours using a metric tree. The computation of a complete similarity matrix is thus not needed, limiting the time needed for distance computations to an average complexity of $O(Nlog_2(N))$ (N is the number of patterns), instead of $O(N^2)$.

This distance is a metric when the expression values are normalized per gene, as mentioned in [16]. Given two expression patterns x and y, it is defined as follows:

$$d(x,y) = \sqrt{1 - CORR(x,y)}$$

Where $CORR(x,y)$ is the Pearson correlation coefficient defined next.

$$CORR(x,y) = \frac{1}{D}\sum_{i=1}^{D}[(x_i - \mu_x)(y_i - \mu_y)] \Bigg/ \sqrt{\frac{\sum_{i=1}^{D}(x_i - \mu_x)^2}{D}}\sqrt{\frac{\sum_{i=1}^{D}(y_i - \mu_y)^2}{D}}$$

Where μ_x, μ_y are the average of expression values of x, and y respectively, and D is the number of dimensions.

A dynamic-range nearest neighbour is defined to capture a compact neighbourhood for a pattern. It is a variation of the static range nearest neighbour used by DBScan, where the dynamic-range depends on the pattern's distance from its nearest neighbour. Given a data set *P*, and a scaling input parameter *f* (*f*>1), the dynamic range of pattern p is defined as:

$$\breve{r}(p) = f \min_{q \in P}\{d(p,q)\} \quad (1)$$

The dynamic range neighbourhood for a pattern *p* is then defined as follows:

$$NN(p) = \{q \mid d(p,q) \le \breve{r}(p)\} \quad (2)$$

This neighbourhood information reflects the structure of data in the vicinity of a pattern, and is used –as shown later- to merge patterns to each other or to other clusters.

B. Clustering Measures

In order to distinguish between different clusters, the clusters' densities are considered, where the distance behaviour is used to reflect those densities. In general, a cluster of relatively high density has smaller distances between its patterns (considering only neighbourhoods of patterns), and a cluster of relatively low density has larger distances between the patterns. The proposed distance-based measures, used to reveal the structure of data, are presented next, and explained later on.

Local Average Distances: which is the average of pattern p's distances from its dynamic-range neighbours. It is defined as follows:

$$\mu_p^{NN} = \frac{\sum_{\forall x \in NN(p)} d(p,x)}{|NN(p)|} \quad (3)$$

Cluster Average Distances: which is the average of distances accepted in a cluster. It is calculated as follows:

$$\mu_c = \frac{\sum_{\forall (p_i,p_j) \in A_c} d(p_i,p_j)}{|A_c|} \quad (4)$$

A_c is the list of associations accepted in a cluster during the clustering process, where an association between two patterns is used to describe a link between them. It is described by the distance and the two patterns' ids as follows: $a_{pq} = (d(p,q), p, q)$

Cluster Harmonic Distance Average: which is the harmonic average of distances accepted in a cluster. It is calculated as follows:

$$\mu_{H_c} = |A_c| / \sum_{\forall (p_i,p_j) \in A_c} 1/d(p_i,p_j) \quad (5)$$

The above measures reflect the distance characteristics of either a pattern's neighbourhood or distances between patterns in a cluster. The local average defined for a pattern reflect the density structure in the vicinity of a pattern, while a cluster's average distances reflect the density of the cluster. The harmonic average, on the other hand, is used for a shrinking process used to get rid of outlier distances (distances extremely larger than normal) in a cluster.

C. Algorithm

The algorithm takes as input the dataset P and 2 main parameters *f* and k. *f* determines the scale by which the neighbourhood of a pattern is decided, and k determines the relative degree of distance consistency at which two clusters can merge.

The algorithm has three main steps, as follows:

1-Get Associations

a-Retrieve dynamic range nearest neighbours for all patterns, and calculate local average distances.

*b-Create associations from patterns' neighbourhoods and order them (ascendingly on distances) in a list **L1**.*

2-Merge Patterns into Clusters

*a-For each association in **L1**, if the merge criterion is satisfied, merge associated patterns/clusters, update the new cluster's average distances, and move association from **L1** to list **L2** (accepted associations' list- initially empty).*

*b-For each association in **L1**, if the enriching criterion is satisfied, merge associated clusters together and move association from **L1** to **L2**.*

3-Refine Clusters

*a-For each association in **L1** that joins two patterns in the same cluster and is consistent to that cluster's average distances, move it to **L2**, and update the average distances of that cluster.*

*b-For each cluster, create its list of associations from **L2**, calculate the harmonic average of its associations' distances, and remove associations not consistent to this measure from the cluster list.*

Given the value of parameter *f*, the nearest neighbours are retrieved for all patterns and arranged in associations, that are sorted into list L1. Following this step, the merging process starts, where initially each pattern is a singleton cluster i.e. a pattern p_i is assigned to a cluster c_i containing only this pattern. Hence, merging two singleton patterns is equivalent to merging two clusters in the proposed merge criterion.

At each step of the merging process, a new association between patterns p1 and p2 in clusters c1 and c2 respectively, is retrieved from L1. Given the value of parameter k, the possibility of merging c1 and c2 is examined using the following merge criterion:

$$d(p_1,p_2) < k.\min(\mu_{c1},\mu_{c2}) \wedge \max(\mu_{c1},\mu_{c2}) < k.\min(\mu_{c1},\mu_{c2})$$

Where $d(p_1,p_2)$ is the association's distance between p1 and p2, and μ_{c1},μ_{c2} are the average distances of c1 and c2 respectively (given in (4)), but local average distances for singleton clusters (as given in (3)). If two clusters are merged, the following changes are done:

-The new cluster's average is updated by the distance value used to merge the clusters (distance belongs to list of accepted associations A_c mentioned earlier in (4)).

-The two merged clusters are given the same label.

-An association satisfying the above criterion is removed from list L1 and stored in list L2.

The above merge criterion demands the existence of a relative consistency between two clusters' average distances, as well as the existence of a relative consistency between the linking-association's distance and each cluster's average distances. This is implemented by bounding one cluster's average distances to that of the other cluster i.e. $(\mu_{c1} < k.\mu_{c2}) \wedge (\mu_{c2} < k.\mu_{c1})$, which corresponds to bounding the maximum value to the minimum one i.e. $\max(\mu_{c1},\mu_{c2}) < k.\min(\mu_{c2},\mu_{c1})$. Also the linking distance should be bounded to both averages, which in result is bounded to the minimum average.

A cluster "enriching" step follows, which uses the rest of associations in L1. An association from L1 is retrieved, connecting two different clusters *s* and *l*, one smaller (*s*) in size than the other (*l*). The two clusters are merged if the following criterion is satisfied:

$$(|s| < \delta.|l|) \wedge (\mu_s < \mu_l) \wedge (d(p_s,p_l) < k.\mu_l)$$

This criterion demands the attraction of large clusters to nearby tiny clusters of size less than δ of the larger cluster's size with two restrictions (to avoid outliers' effect):

-The tiny cluster should be denser (smaller average distances) than the larger one in order to avoid attracting outliers.

-The associating distance should be consistent to the major cluster's average distances.

δ is substituted in all the experiments by 5%, which is chosen small enough to avoid violating the main merging criterion. During this process, the clusters maintain their original distance averages obtained during merging, and only the tiny cluster is given the same label as the larger one. The accepted merging association is added to list L2 and removed from L1.

A refining process follows the merging process, where weak associations are removed from each cluster, which may result in breaking a cluster into two or more new clusters. Prior to cluster refining, any association from L1 that connects two patterns of the same cluster *c* is added to the list of accepted associations L2 if its distance satisfies the condition $d(p_1,p_2) < k.\mu_c$. The affected clusters' average is updated accordingly. This is done to ensure the inclusion of all internal associations consistent to the cluster's average distances, in the cluster's list of accepted associations.

Associations that belong to each cluster are then arranged into lists using L2. The harmonic average of associations' distances is calculated for each cluster as given in (5), and associations of distances not conforming to this average-as given next- are removed from the cluster:

$$d(p_1,p_2) < k.\mu_{H_c}$$

Where μ_{H_c} is defined in (5). This method is used as a way of shrinking the cluster average distances towards the denser core of the cluster, enabling the identification and removal of outlier distances from the cluster.When weak associations are removed, patterns in each cluster are re-labeled, and the final clusters result.

Any singleton patterns or patterns in clusters of size less than 1% of the total data size are considered outliers, and are allocated to clusters of majority in their neighborhoods.

D. Performance Issues

The time complexity of Mitosis is $O(DN\log_2(N))$, where D is the number of dimensions, and N is the number of patterns. This is the same complexity of DBScan. Mitosis, however performs a number of scans on the neighbourhood associations, which are not done by DBScan due to its simpler solution. Yet, Mitosis is able to discover clusters of arbitrary shapes and of different densities, not discovered by DBScan due to its prespecified static density.

Compared to other related algorithms, Mitosis is faster as it uses nearest neighbour information rather than a similarity matrix. CLICK and HCS, on the other hand need a similarity matrix as an input to the algorithm. This elevates the time complexity to a quadratic one in terms of the number of patterns, i.e. $O(DN^2)$.

SPIN is of $O(DN^2+N^3)$ complexity due to similarity matrix calculation and permutations.

IV. EXPERIMENTS

Both time series and nontemporal gene expression data sets have been used to examine the efficiency of the proposed algorithm. The results were compared to results obtained by CLICK (obtained from [26]), DBScan[13], and K Means. DBScan is not known for application in high dimensional gene expression analysis. Only in [17], do the authors refer to using Optics[23] , which depends on DBScan. Yet the authors didn't consider its importance in finding connected patterns. Results are also assisted by results from SPIN [7], and results from Hotler et. al [8] who used SVD to explore the data sets. The data sets used are: Serum gene expression time series [9], and Leukaemia data set [11]. The first data set illustrates the ability of Mitosis to identify connected patterns, and the second illustrates its ability to identify clusters of different densities, revealing important information in data.

Colour gradient representation, similarity matrix, and SVD are used to visualize the results. SVD have been used for visualizing gene expression data in [8], and [20]. The two highest ranked modes are selected and their coefficients are plotted against each other in a 2-D plot. It is used to reveal some aspects of the data sets, yet this projection hides other information. It is used here only for illustration, and it is not part of the clustering algorithm. The colour gradient representation is used to view the gene expression patterns, where darker colours are used to reflect higher expressions (up regulations) and brighter ones reflect the lower expressions (down regulations).

Aside from the general preprocessing for handling missing values, row normalization is done for each gene in order to be able to use the Pearson-based metric discussed above. The row normalization considered here is the mean-standard deviation normalization.

Parameters are selected by detecting the stability in the k/cluster curve at each f value. For a given f value, the k/cluster curve plots the number of clusters, obtained from the clustering solutions at all k values, against k. Parameter f is normally selected starting at values slightly greater than 1, and is increased in small steps. The values of k start at values above 1, and have a maximum value bounded by finding the minimum number of clusters attainable by a certain f value. The stability in the k/cluster curve is detected for all consecutive values of f. When a consistency between a number of consecutive f values, with respect to their stability is achieved, the value of k corresponding to the stability at a particular f value is selected (see figure 6 for selecting parameters for the serum dataset).

A. Time Series Gene Expression :Serum Dataset

Serum data set is obtained from [9] and contains 12 time point expressions for about 500 genes. The color-gradient representation, similarity matrix (using visualization from [26]) and SVD modes of serum's dataset reveal the continuity in gene expression along the whole data set. The color-gradient in fig. 3 (left figure) shows a gradual transition of expression, while the similarity matrix (right figure) shows a matrix arrangement similar to that given by SPIN for a 2-D circular connected shape.

DBScan was examined for a large range of *Eps* and *Minpts* settings. Settings of *Eps* and *Minpts* at (0.4,20), (0.5,50) and (0.6,90) gave one connected cluster (fig. 4-d from left to right). Given the *Eps* settings of 0.4, 0.5 and 0.6, the larger group of settings for *Minpts* gave the same connected cluster, while only few settings gave two clusters, including an unstable cluster.

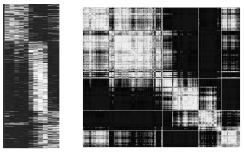
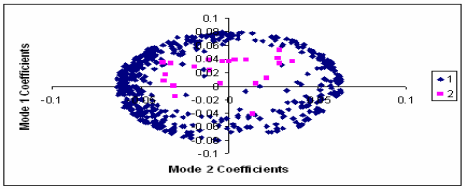
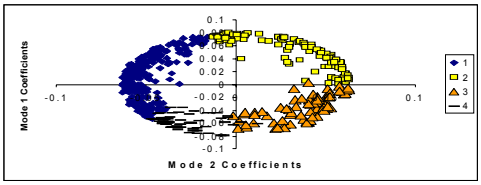


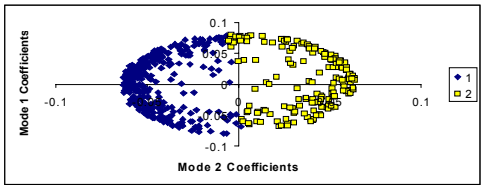
Figure 3: Serum color gradient representation (left) and similarity matrix (right). The brightest black/white color stands for the highest/lowest expression values in the color gradient and the largest/smallest distances in the similarity matrix.



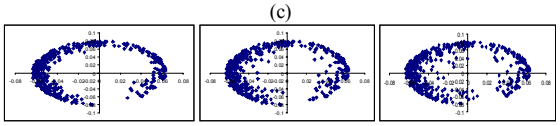
(a)



(b)



(c)



(d)

Figure 4: Clusters obtained for Serum data set by (a)Mitosis, (b)CLICK , (c)K Means and (d)DBScan, SVD modes' coefficients are plotted for visualization.

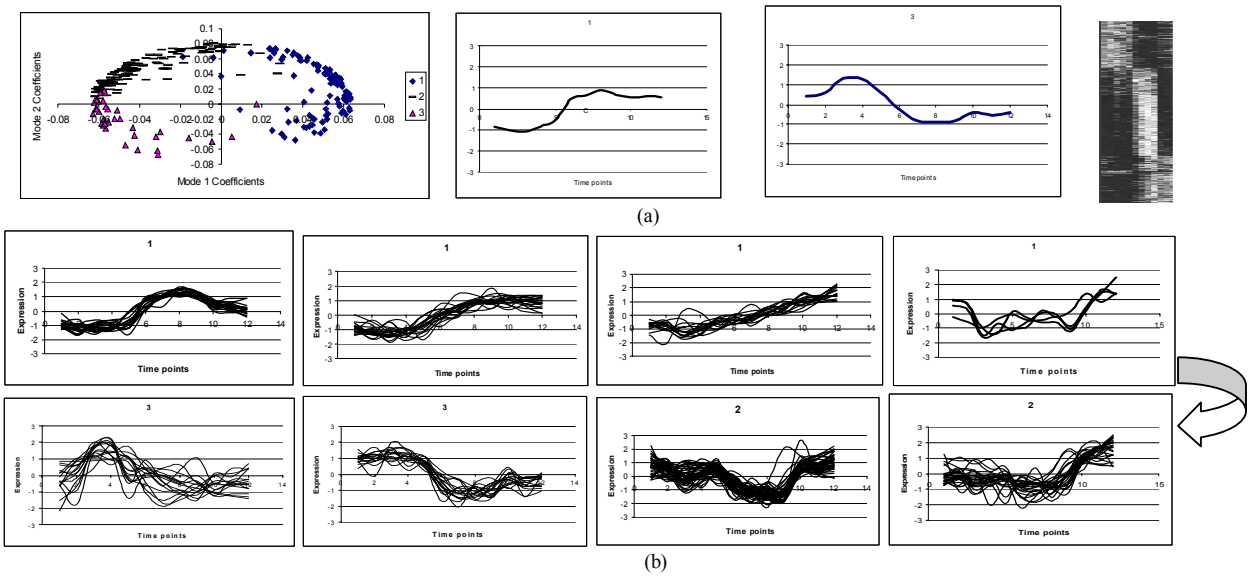


Figure 5: Illustration of gradual change in patterns of serum data set, (a) SVD modes of 3 discrete groups (1,2,3), corresponding mean expression patterns of two distant clusters (1,3), and the color gradient of the main cluster discovered by Mitosis (b) Gradual change in patterns between the means of (1,3) (clockwise order starting from 1 till reaching 3)

Running CLICK using its default homogeneity discovered 4 clusters (fig. 4-b) of coherent patterns, and left out a number of patterns unclassified (singletons).When decreasing the homogeneity setting to 0.1, two clusters resulted ,similar to those obtained by K Means when setting the number of clusters to 2 (fig. 4-c). Note that CLICK uses a validity based on homogeneity-separation, which prefers coherent patterns.

Mitosis, after tuning its parameters, results in one cluster containing most of the genes, and a very small cluster (fig.4-a) at a parameter setting of f=1.3 and k=1.5. The choice of parameters is illustrated in figure 6, where the k/cluster curves at f values of 1.1, 1.2 and 1.3 are shown (clusters of size less than 1% of the data size were ignored). The common stability of the 3 curves is at a number of clusters=2. The highest f value of 1.3 was selected, and the corresponding k value of 1.5 that gave 2 clusters is selected. This result reveals the connectivity of expression patterns observed in the similarity matrix, SVD, and the color gradient representation. To further illustrate the gradual change of expression patterns in the cluster discovered by Mitosis, fig. 5.a shows the SVD representation for three discrete clusters (1,2,3) that together cover the main cluster discovered. The gradual change of patterns between two clusters (1,3 in fig. 5.a), that seem opposite in regulatory pattern is shown in fig. 5.b. It can be observed that the up-regulation in the first five time points (1-5) for cluster 1 is shifted to the next five time points (5-10) for cluster 3. Similarly, the down-regulation in the first five time points for cluster 3 is shifted to the next five time points for cluster 1. The color-gradient confirms the same gradual change between the two distant clusters. Joining the three discrete clusters in one cluster as done by Mitosis, reveals the connectivity between them through this gradual change, and the possibility of finding regulatory relations between genes. Results obtained by the use of SVD in Hotler et. al [8], were interpreted as continuity in gene expression patterns. The authors comment that "..the progressive changes in gene expression are uniform and continuous. Thus, genes are generally not activated in discrete groups or blocks….".

Unfortunately there is no common ground truth for classifying this data, and comparisons rely on observations found in the literature as that of Hotler et. al [8].

For this data set, Mitosis and DBScan gave almost the same results, revealing one connected cluster in the data set. This data set illustrates the ability of Mitosis to obtain connected patterns of gene expressions. Further investigation of genes in the main connected cluster, could be assisted by other gene expression arrays to help build gene regulatory networks.

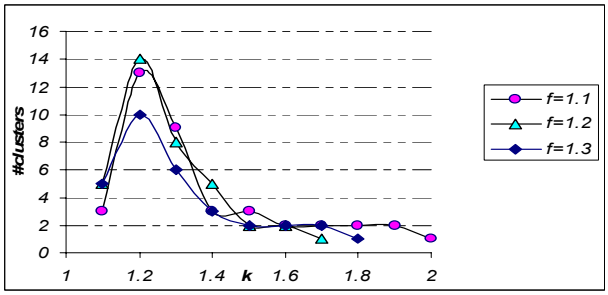


Figure 6: Parameter selection for serum data set, and detecting stability at number of clusters=2

B. Leukaemia dataset

Leukemia dataset [11] is an example of a non temporal gene expression set. It is used to illustrate the ability of Mitosis to discover connected patterns of arbitrary densities. The data set

contains the expression of 999 genes along 38 samples obtained from ALL (27 samples) and AML (11 samples). Furthermore, the ALL samples are arranged in 18 B lineage and 9 T lineage samples. The order of samples along the data set columns is: ALL B lineage, ALL T lineage and AML.

In previous studies of this data set [3], clustering resulted in 3 groups of expressions linked to ALL B lineage, ALL T lineage and AML. While SPIN [7] discovered that ALL B lineage and ALL T lineage gene expressions are connected in one elongated cluster. The SVD representation is given in fig. 7.

Mitosis at the parameter setting of f=1.4 and k=1.08 finds 2 dense clusters (fig. 8-a), which correspond to ALL and AML groups, and also discovers a third cluster connecting patterns from ALL and AML (fig. 8-b). This cluster of expressions connecting both classes is of lower density than those of ALL and AML clusters. This third group of genes was not identified by any of DBScan, CLICK, or K-means. The SVD presentation (fig. 7) shows the continuity of gene expression among the two main clusters of ALL and AML. The first color-gradient representation in fig. 8-c shows the ALL cluster of gene expressions, with a high expression in the 18 left most columns corresponding to ALL B lineage samples, and a high expression in the following 9 columns, corresponding to those expressed in ALL T lineage samples. It is obvious that some genes are expressed in both T lineage and B lineage samples, as apparent in the faint black streaks appearing in the first 18 columns (ALL B) accompanying the high expression in the next 9 columns (ALL T).

The AML cluster is represented by the middle color-gradient of fig. 8-c, where expression in the last 11 columns corresponding to AML samples is obvious. The right most color-gradient of fig. 8-c represents the third low dense cluster. Genes expressed for both ALL B and ALL T lineages, and others expressed for both ALL T lineage and AML are apparent. Genes that are also expressed in both of ALL B lineage and AML are also present in this cluster, as will be discussed later.

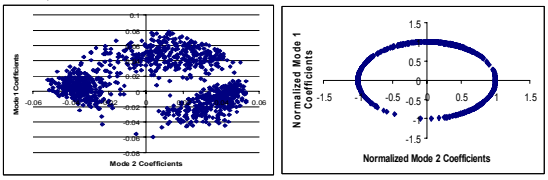


Figure 7: Coefficients of SVD modes for the leukemia data set (left figure), and normalized coefficients (right figure).

DBScan finds two clusters, as shown in the left figure of fig. 9, for a large number of *Eps,* and *Minpts* settings: 0.5,2 to 0.5,22 and 0.6,50 to 0.6,80. These clusters correspond to ALL and AML classes, while at the setting of 0.5,24 it finds three weak clusters (fig. 9, right figure). However DBScan fails to find the low dense cluster found by Mitosis. CLICK, at the default homogeneity, finds three main clusters (fig. 10, left figure), and a fourth tiny cluster of 23 genes. The main classes correspond to ALL B, ALL T, and AML. The fourth one is in between ALL T and AML. When lowering the homogeneity to 0.45, it gives 6 clusters: 3 large ones, and 3 tiny ones. When lowering the value to 0.2 (fig. 10, right figure), two clusters result corresponding to ALL and AML, and in that case, resembling K means results found at k=2 (fig. 11, right figure). In both cases of low and high homogeneity CLICK failed to find the low dense cluster found by Mitosis.

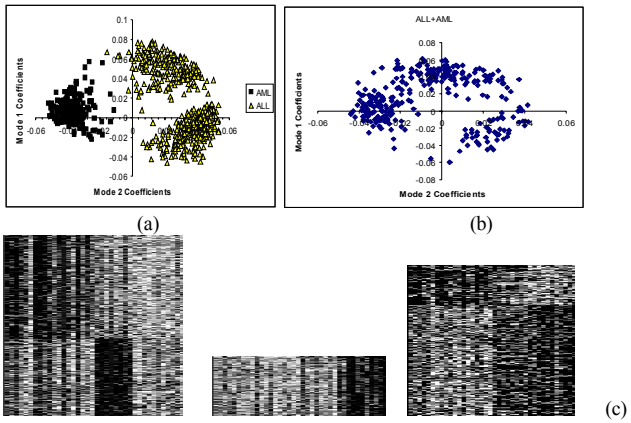


Figure 8: Mitosis results for leukemia data set, (a) Two dense clusters of ALL and AML, (b) A lower density cluster combining genes from ALL and AML, (c) Color-gradient representation of ALL cluster (left), AML cluster(middle) and ALL+AML cluster (right) .

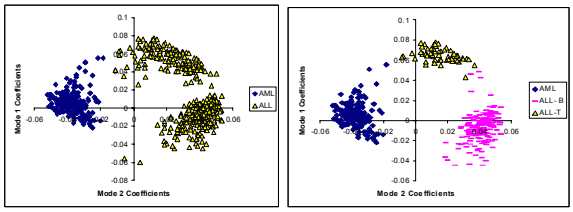


Figure 9: DBScan clusters for Leukemia data set, at Eps=0.6, Minpts=60 (left figure), and Eps=0.5, Minpts=24 (right figure)

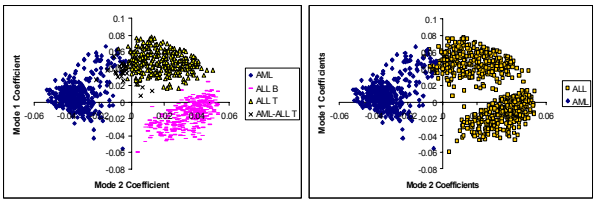


Figure 10: CLICK results for Leukemia data set, at the default homogeneity (left figure) and at homogeneity =0.2 (right figure)

Biological Implications:

Genes in the mixture of ALL and AML cluster found by Mitosis, include HOXA9, SM22 Alpha identified by [21] as belonging to MLL (Mixed Lineage Leukemia) group. In [15], the authors state that HOXA9 gene may hold an important key to MLL Leukemia as it is one homeobox gene most frequently overexpressed in

MLL Leukemia. Also HoxB2 and HoxB3 genes were included in that third cluster. The expression patterns for HOXA9, SM22 Alpha, HOXB2, and HOXB3 are shown in fig. 12, illustrating that HOXA9 is expressed for ALL B lineage and AML samples, SM22 Alpha is expressed for ALL T lineage and AML samples, HOXB2 is expressed for ALL B and T lineages and AML samples comparably, and HOXB3 is expressed for ALL B lineage and AML samples. It is concluded that genes belonging to this third cluster, and that are expressed both in AML and ALL should further be studied for investigating their relation to MLL, or other types.

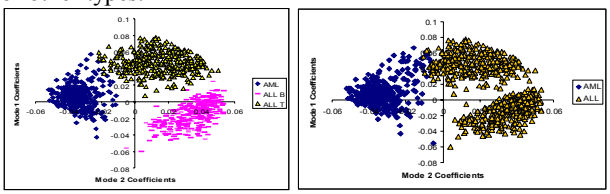


Figure 11: K means results for Leukemia data set, at k=3 (left figure) and k=2 (right figure).

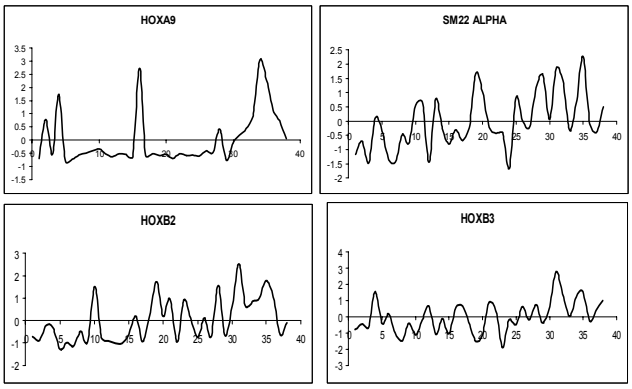


Figure 12: Gene expressions of selected genes from the third cluster discovered by Mitosis for leukemia data set.

V. CONCLUSION

An algorithm for finding connected expression patterns in gene expression data sets is proposed. The algorithm depends on using nearest neighbours and thus is able to find connected instead of coherent patterns. It has a benefit over known density based clustering, in its ability to discover clusters of different densities by using novel measures. Its efficiency in discovering connectivity is illustrated using both time series data set and non temporal data set. The continuity of expression in time series data is important in discovering genetic pathways, while connected patterns in non temporal data discovers new clusters that gather properties from different sample types.

REFERENCES

[1] A. Ben-Dor, R. Shamir, and Z. Yakhini, "Clustering gene expression patterns.", *Journal of Computational Biology*, 6(3/4):281–297, 1999.

[2] R. Shamir, and R. Sharan, "Algorithmic Approaches to Clustering Gene Expression Data.", in *Current Topics in Computational Biology*, MIT Press, pp. 269-299 (2002).

[3] S. Monti , P. Tamayo , J. Mesirov ,and T. Golub, "Consensus Clustering:A resampling-based method for class discovery and visualization of gene expression microarray data",© 2003 Kluwer Academic Publishers.

[4] T. Golub., et. al, "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring", in *Science* Vol. 286. no. 5439, pp. 531 – 537, (1999).

[5] P. D'haeseleer, S. Liang., and R. Somogyi, "Genetic Network Inference:From Co-Expression Clustering to Reverse Engineering", *Bioinformatics* 16(8):707-26, (2000).

[6] D.W. Mount, *Bioinformatics, Sequence and Genome Analysis,* Cold Spring Harbor laboratory Press, NY, USA, 2001.

[7] D. Tsafrir , I. Tsafrir ,L. Ein-Dor., O. Zuk, D.A. Notterman and E. Domany., "Sorting points into neighborhoods (SPIN): data analysis and visualization by ordering distance matrices", in *Bioinformatics 2005 21(10)*.

[8] N.S. Holter, M. Mitra , A. Maritan , M. Cieplak, J.R. Banavar , and N.V. Fedoroff , "Fundamental Patterns Underlying Gene Expression Profiles: Simplicity from Complexity.", in *PNAS, July 18, 2000, vol. 97, no.15.*

[9] www.sciencemag.org/feature/data/984559.shl

[10] V.R. Iyer et. al. "The Transcriptional Program in the Response of Human Fibroblasts to Serum", in *Science (283), 83–87, 1999*.

[11] http://www.broad.mit.edu/cgi-bin/cancer/datasets.cgi.

[12] M.B. Eisen, P.T. Spellman, P.O. Brown. and D. Botstein, "Cluster Analysis and Display of Genome-wide Expression Patterns." ,*PNAS Vol. 95, 14863-14868, December 1998*.

[13] M. Ester ,H-P Kriegel, J. Sander, and X. Xu , "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", Published in *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*.

[14] G. Karypis, E. Hong, and H.V. Kumar., "CHAMELEON: A Hierarchical Clustering AlgorithmUsing Dynamic Modeling",*University of Minnesota, Technical Report #99–007*.

[15] A.R. Kumar , W.A. Hudson, W. Chen , R. Nishiuchi,,Q Yao , and J.H. Kersey , "*Hoxa9* influences the phenotype but not the incidence of *Mll-AF9* fusion gene leukemia", *Blood,* 1 March 2004, Vol. 103, No. 5, pp. 1823-1828.

[16] J.A. Hartigan, "Clustering Algorithms", *Wiley Series in Probability and Mathematical Statistics*, 1975.

[17] D. Jiang , J. Pei , and A. Zhang., "An Interactive Approach to Mining Gene Expression Data" , *IEEE Trans. on Knowledge and Data Engineering*, vol. 17, no. 10, pp. 1363-1378, Oct., 2005.

[18] http://www.pami.uwaterloo.ca/~nyousri/DynamicModel.pdf

[19] E. Hartuv, and R. Shamir, "A clustering algorithm based on graph connectivity". *Information Processing Letters*, 76(200):175{181, 2000.

[20] O. Alter , O.B.Patrick , and D. Botstein ,"Singular value decomposition for genome-wide expression data processing and modeling", *Proc. Natl. Acad. Sci. USA*, 97(18), August 2000.

[21]S.A. Armstrong, et al. "MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia". *Nat Genet*. 2002;30: 41-47.

[22] C. Sheikholeslami, S. Chatterjee, and A. Zhang, , "WaveCluster: A-MultiResolution Clustering Approach for Very Large Spatial Data set". *Proc. of 24th VLDB Conference*. .(1998).

[23] M. Ankerst, M. Breunig, H-P Kriegel, and J. Sander., "OPTICS: Ordering Points to Identify the Clustering Structure", *Proc. ACM SIGMOD'99 Int. Conf. on Management of Data.*

[24] A. Hinneburg and D. Keim, "An efficient approach to clustering in large multimedia data sets with noise", *In Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining*, pages 58-65 (1998).

[25] P. Ciaccia, M. Patella, and P. Zezula. "Mtree:An efficient access method for similarity search in metric spaces." In *Proceedings of the 23rd Conference on Very Large Databases* (*VLDB'97*), pp.426–435.

[26] http://www.math.tau.ac.il/_rshamir/click/click.html