ISSN 22773061

A Preview on Subspace Clustering of High Dimensional Data

Sajid Nagi[1], Dhruba K. Bhattacharyya[2], Jugal K. Kalita[3]
[1]Department of Computer Science, St. Edmund's College, Shillong – 793001.
sajidnagi@gmail.com
[2]Department of Computer Science and Engineering, Tezpur University, Napaam – 784028.
dkb@tezu.ernet.in
[3]Department of Computer Science, University of Colorado, Colorado Springs CO 80918, USA.
jkalita@uccs.edu

Abstract: When clustering high dimensional data, traditional clustering methods are found to be lacking since they consider all of the dimensions of the dataset in discovering clusters whereas only some of the dimensions are relevant. This may give rise to subspaces within the dataset where clusters may be found. Using feature selection, we can remove irrelevant and redundant dimensions by analyzing the entire dataset. The problem of automatically identifying clusters that exist in multiple and maybe overlapping subspaces of high dimensional data, allowing better clustering of the data points, is known as Subspace Clustering. There are two major approaches to subspace clustering based on search strategy. Top-down algorithms find an initial clustering in the full set of dimensions and evaluate the subspaces of each cluster, iteratively improving the results. Bottom-up approaches start from finding low dimensional dense regions, and then use them to form clusters. Based on a survey on subspace clustering, we identify the challenges and issues involved with clustering gene expression data.

Keywords: Challenges, clustering survey, gene expression data, high dimensional data, issues, subspace clustering

Council for Innovative Research
Peer Review Research Publishing System
Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY
Vol 6, No 3
editor@cirworld.com
www.cirworld.com, member.cirworld.com

1. INTRODUCTION

The purpose of cluster analysis is to detect groups or clusters of similar objects, where an object is represented as a vector of measurements or points in multidimensional space. The distance measure determines the dissimilarity between objects in the various dimensions in the dataset [1]. With advances in technology, data collection has become easier and faster, leading to large and complex datasets containing many objects and dimensions. This requires that the existing algorithms be enhanced to speedily detect clusters of high quality. In sharp contrast to subspace clustering algorithms, traditional clustering algorithms consider all of the dimensions of the dataset in discovering clusters although many of the dimensions are often irrelevant [2][3][4]. These irrelevant dimensions can confuse clustering algorithms by hiding clusters in noisy data. The purpose of generating high quality clusters in high dimensional data such as microarray gene data is to get a correct and informative biological interpretation of the gene cluster. One such technique to extract biologically relevant information about genes in a dataset is a tree of genes called GERC tree [5], which is produced by a divisive clustering algorithm, and the leaves represent the generated clusters.

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 under 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. A method is presented in [6] to build a gene co-expression network (CEN), which is an undirected graph of nodes representing genes, connected by an edge if the corresponding gene pairs are significantly co-expressed. A gene expression similarity measure called NMRS (Normalized mean residue similarity) is used to construct the CEN, which is used to detect network modules from the built network.

1.1. Problems associated with Clustering High Dimensional Data

In high dimensional data, it is common for all the objects in a dataset to be spread out until they are almost equidistant from each other. The distance measures between the objects become meaningless and because of this “curse of dimensionality” [7], the performance of many clustering algorithms suffer, giving rise to several issues. (i) Any optimization problem becomes increasingly difficult with an increasing number of variables (attributes) [8]. (ii) The discrimination between the nearest and the farthest neighbors becomes rather poor in high dimensional data spaces [9][10]. (iii) Many irrelevant attributes in a data set are collected due to the highly automated data acquisition process. Since the clusters are defined by some of the attributes only, the remaining irrelevant attributes (“noise”) may interfere with the efforts of finding the “true” number of clusters. (iv) Also, in a data set containing many attributes, some attributes will most likely exhibit correlations among them. (v) The wrong selection of a proximity measure used by a clustering technique may lead to the discovery of some similar groups of genes at the expense of obscuring other similar groups.

Feature selection methods have been employed somewhat successfully to improve cluster quality. These algorithms find a subset of dimensions on which to perform clustering by removing irrelevant and redundant dimensions. Unlike feature selection methods which examine the dataset as a whole, subspace clustering algorithms evaluate features only on a subset of the data, based on a measure referred to as a “measure of locality” [11] representing a cluster, and are able to uncover clusters that exist in multiple, possibly overlapping subspaces and represent them in easily interpretable and meaningful ways [12].

1.2. Relevance of Subspace Clustering

Subspace clustering has been applied in text-mining, network anomaly detection, object detection in hyper-spectral satellite data and gene expression data analysis for finding co-expressed or coherent patterns. Clustering algorithms have been used with DNA microarray data for identification and characterization of genes. However, the high dimensionality of microarray and text data makes the task of pattern discovery difficult for traditional clustering algorithms and for this reason subspace clustering techniques can be used to uncover the complex relationships found in data in these areas. The work proposed in [6] has been extended to trace correlation among genes over a subspace of samples, represented by a co-expression network [13].

1.3. Proximity Measures

There are different methods [14] for quantifying similarity or dissimilarity between two 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., $D(g_i, g_j) = D(g_j, g_i)$.
v. The distance measure obeys the triangle inequality property, i.e., for profiles g_i, g_j, g_k, we have $D(g_i, g_k) \leq D(g_i, g_j) + D(g_j, g_k)$.

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. The choice of the proximity measure depends upon the type of data, dimensionality and the approach used in the identification of coherent patterns. Therefore, to identify genes or samples that have similar expression profiles, selection of an appropriate proximity measure is very essential.

2. GENE-BASED CLUSTERING USING SUBSPACE APPROACHES

The purpose of gene-based clustering is to group together co-expressed genes indicating co-function and co-regulation. It has already been established in molecular biology that normally only a small subset of genes participate in any cellular process of interest. However, traditional clustering algorithms are generally concerned with clusters in the full feature space. Subspace clustering was initially proposed by Agrawal et al. [11], to evaluate features in only a subset of the data, based on a “measure of locality” representing a cluster. Sub-space clustering algorithms are further divided into two categories (a) *bottom-up search* and (ii) *top-down search*.

2.1. Bottom-Up Subspace Search Methods

This category of methods takes advantage of the downward closure of the property of density to reduce the search space. This property states that *if a subspace S contains a cluster, then any subspace T* ⊆ *S must also contain a cluster*. It determines locality by creating bins (for each dimension) which finally form multi-dimensional grid. There are two approaches: (i) static grid-sized approach and (ii) data driven strategies adopted to determine the cut-points. For the first approach, the two popular algorithms of this category are CLIQUE [11] that attempts to find clusters within subspaces using a grid-density based clustering approach and ENCLUS [14], which is an *apriori* like clustering method that defines clusters based on entropy. For the second approach, the algorithm MAFIA [15], a variant of CLIQUE, uses an adaptive, grid-based approach with parallelism, to improve scalability. Unlike CLIQUE and MAFIA, CBF [16] uses an efficient algorithm for creation of partitions optimally, to avoid exponential growth of bins with the increase in the number of dimensions. CLTree [17] adopts an algorithm which separates high and low density areas by using a modified decision tree algorithm that uses a modified gain criterion to measure the density of a region. DOC [18] is a hybridization of bottom-up and top-down approaches and introduces the concept of an optimal projective cluster with strong clustering tendency over a subset of dimensions by an iterative improvement pattern.

The algorithms under the bottom-up approach are suitable for clustering gene expression data as they are (i) able to handle high dimensional data, (ii) can find clusters of arbitrary sizes and shapes, (iii) can find clusters which are embedded, intersected or disjoint, and (iv) scale reasonably well with the amount of data. However, the disadvantages of these algorithms are that (i) the algorithms do not scale well with the increase in number of dimensions, (ii) the algorithms may sometime eliminate small clusters, and (iii) the running time grows exponentially with the increase in the number of dimensions in the datasets.

2.2. Top-Down Subspace Search Methods

This approach starts with an initial approximation of clusters in an equally weighted full feature space. Next, it follows an iterative procedure to update the weights and accordingly reforms the clusters. It is expensive in the full feature space. However, the use of sampling techniques can improve the performance. The number of clusters and the size of the subspace are the most critical factors in this approach. Several algorithms have been described in the literature such as PROCLUS [19], which is a sampling based top-down subspace clustering algorithm that randomly selects a set of *k*-medoids from a sample and iteratively improves the choice of medoids to form better clusters. ORCLUS [20], like PROCLUS, also attempts to form clusters iteratively by assigning a point to its nearest cluster. It computes the dissimilarity between a pair of points as a set of ortho-normal vectors over a subspace. FINDIT [21] is a sampling based subspace clustering algorithm that finds clusters in a three phased manner. The algorithm δ-Clusters [22] starts with an initial seed and attempts to improve the overall quality of the clusters iteratively by swapping dimensions with instances. The use of coherence as a similarity measure makes it more relevant for microarray data analysis. COSA [23] uses *k*nn to iteratively calculate the dimension weight for each instance and assigns higher weighted dimensions to those instances which have less dispersion within the neighbourhood till the weights stabilize. The output of COSA is a distance matrix, which can be used as an input to any distance based clustering algorithm.

Top-down subspace search methods are fast and scalable and the performance improves if sampling is used for large databases. However, its main disadvantages are that (i) it is sensitive to input parameters, (ii) quality of clusters depends upon the size of the sample chosen, and (iii) sampling may lead to some significant results being missed.

2.3. Biclustering Algorithms

A bicluster [24] is an *I* x *J* sub-matrix that exhibits some coherent tendency where *I* and *J* are the genes (rows) and conditions (columns) respectively, and $|I| \le |N|$ and $|J| \le |M|$. A biclustering algorithm introduces a measure for the residue, called *mean squared residue*, which is an indicator of the degree of coherence of an element with respect to the remaining elements for the particular given bicluster. The lower the mean squared residue, the stronger is the coherence exhibited by the cluster and the better is the quality of the bicluster. The problem of finding the largest bicluster with minimum mean squared residue is NP-hard [24]. Biclustering algorithms employ different heuristic approaches to address this problem and can be divided into the following categories [25].

2.3.1 Greedy Iterative Search

Greedy Iterative search is based on the idea of forming biclusters of rows/columns by addition or deletion, with an attempt to maximize the local gain, which may lead to faster processing at the cost of losing good biclusters. Cheng and Church [24] pioneered the application of the greedy approach to gene expression data with the limitation that overlapping/embedded clusters cannot be identified because the elements of the already identified biclusters are masked by random noise. This limitation was addressed by FLOC [26] which uses a probabilistic algorithm to discover a set of *k*-possible overlapping biclusters simultaneously. OPSM [27] is another probabilistic model that attempts to address the idea of large Order-Preserving SubMatrices (OPSM) with maximum statistical significance, where a bicluster is determined by a set of rows, a set of columns and a linear ordering of the columns. Murali and Kasif proposed xMOTIF [28]. Its purpose is to compute the set of conserved rows *I* and the set of columns *J* that give the largest xMotif, i.e., the one that contains the largest number of rows.

2.3.2 Divide-and-Conquer

Divide and conquer algorithms have the significant advantage of being potentially very fast. However, they have a significant drawback of missing good biclusters that may be split before they can be identified. The *Block Clustering* [29] algorithm begins with the entire data in one block (bicluster) and iteratively tries to find the best split. Since the estimation of the optimal number of splicings is difficult, Duffy and Quiroz [30] suggested the use of permutation tests to determine when a given block split is not significant. Following this direction, Tibshirani et al. [31] added a backward pruning method to the block splitting algorithm and designed a permutation-based method called *Gap Statistics*, to induce the optimal number of biclusters, *K*.

2.3.3 Exhaustive Bicluster Enumeration

Exhaustive bicluster enumeration methods are based on the idea that the best biclusters can only be identified using an exhaustive enumeration of all possible biclusters exist in the data matrix. These algorithms can certainly find the best biclusters, if they exist, but have a very serious drawback. Due to their high complexity, they can only be executed by assuming restrictions on the size of the biclusters. Tanay et al. [32] introduced SAMBA, a bi-clustering algorithm that performs simultaneous bicluster identification by using exhaustive enumeration.

2.3.4 Iterative Row and Column Clustering Combination

This method applies clustering methods on the columns and rows of a data matrix and then combines the results to obtain biclusters. CTWC [33] tries to identify couples of small subsets of features (F*j*) and objects (O*j*), where both F*j* and O*j* can be either rows or columns. ITWC [34] is an iterative biclustering algorithm based on a combination of the results obtained by clustering performed on each of the two dimensions of the data matrix separately. DCC [35] uses self-organizing maps (SOM) to perform clustering in the row and column spaces of the data matrix and uses angle-metric as similarity measure.

2.4. TriClustering Algorithms

TriClusters are coherent clusters along gene-sample-time (temporal) or gene-sample-region (spatial) dimensions, which may be arbitrarily positioned and overlapped [36]. TriClustering algorithms are used for mining such coherent clusters in three-dimensional gene expression datasets. TriCluster [36] uses a graph-based approach to detect different types of clusters depending on different parameter values, including arbitrarily positioned and overlapping clusters. gTRICLUTER [37] accepts four input parameters, namely, minimum similarity threshold, minimum sample threshold, minimum gene threshold and minimum time threshold and gives as output coherent clusters along gene-sample-time dimension. [38] uses a heuristic TRI-Clustering algorithm to integrate gene expression and gene regulation information, by defining regulated expression values (REV) as indicators of how a gene is regulated by a specific factor. A selected survey outlining the basic challenges of triclustering are presented in [39], based on an analysis of three popular triclustering algorithms. [40] proposes a technique, based on order preserving submatrices, to find a set of triclusters from gene-sample-time data.

3. CLUSTER VALIDITY MEASURES

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 different runs of a single clustering algorithm using different parameters, generally produce different sets of clusters [41]. Therefore, it is important to compare various clustering results and select the one that best fits the “true” data distribution. Cluster validation assesses the quality and reliability of the clusters obtained 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, while objects in different clusters are dissimilar with each other. The second aspect relies on a given “ground truth” of the clusters. The “ground truth” could come from domain knowledge, such as known function families of genes or from other sources such as the clinical diagnosis of normal or cancerous tissues. Cluster validation is based on the agreement between clusters obtained and the “ground truth.” The third aspect of cluster validity focuses on the reliability of the clusters or the likelihood that the cluster structure is not formed by chance. Some of the popular cluster validity measures used to compare clustering results are Rand index [42], Cluster Homogeneity [43], Silhouette index [44], Z-score [45] and P-values [46].

4. CHALLENGES IN SUBSPACE CLUSTERING OF 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.

4.1. Research Challenges

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.

i. The effectiveness of a clustering technique is highly dependent on the proximity measure used by the technique. Finding an appropriate proximity measure or developing a clustering technique which works independently of any proximity measure is a challenging task.

ii. Most existing clustering techniques are either dependent on input parameter(s) or stopping criteria for discovery of the “true” number of clusters. However, providing an appropriate set of parameter(s), or stopping criteria, or developing a parameterless clustering technique able to find biologically relevant clusters is a major task and hence a challenge.

iii. Gene expression data often contain clusters which are “highly connected” [47], intersected or even embedded [48]. Hence, the clustering algorithm should be capable of identifying all these types of clusters simultaneously. This is a challenging task, irrespective of size of dataset or dimensionality.

iv. 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.

v. Apart from clustering, an algorithm should be capable of showing the associations among the clusters which may be useful for drawing conclusions, i.e., the clusters to be represented in interpretable and meaningful ways.

vi. The problem for subspace algorithms is compounded in that they must also determine the appropriate dimensionality of the subspaces.

vii. Subspace clustering algorithms must also define the subset of features that are relevant for each cluster and these are in most cases found to be overlapping.

viii. In addition to producing high quality, interpretable clusters, subspace clustering methods must also be scalable with respect to the dimensionality of the subspaces where the clusters are found. In high dimensional data, the number of possible subspaces is huge, requiring efficient search algorithms.

4.2. Issues to be Addressed

Based on our survey, we identify the following issues.

i. *Proximity Selection*: Concepts such as proximity, distance, or neighborhood become less meaningful with increasing dimensionality of a dataset [9][10][49]. That is, discrimination between the nearest and the farthest neighbors becomes rather poor in high-dimensional space. Euclidean distance, Pearson correlation and cosine angle all seem to work reasonably well as distance measures [45] and [50]. Euclidean distance seems to be more appropriate for ratio data, whereas Pearson correlation seems to work better for absolute-valued data [45].

As a solution, a more deliberate choice of distance metrics (e.g., the use of Manhattan distance or even fractional distance metrics) has been proposed in [49]. Another method is to normalize and standardize the expression profile of each gene [51], with an aim to filter out genes with a flat profile by detecting differences between replicates and separating genes which are not significantly different from the rest. Still another proposed approach [52] is based on inferring confidence intervals, making a more efficient use of the measured data and avoiding the subjective choice of a dissimilarity measure.

ii. *Missing Value*: The acquisition and analysis of microarray data influence the interpretation of the results. It can lead to erroneous conclusions about the data and substitution of missing values may introduce inaccuracies and inconsistencies. So accurate prediction of missing values remains an important issue.

iii. *Relevance to Biologists*: Appropriate clustering can reveal hidden structures in biological data and it can provide accurate means for extracting biologically significant pattern(s). It is particularly helpful to biologists in investigating and understanding the activities of uncharacterized genes and proteins.

iv. *Cluster Expansion*: While expanding a cluster, one must consider cluster quality along with cluster validity. Cluster validity should not be an overhead but should continue to be high simultaneously alongside high quality, so that one can save on time complexity.

v. *Fusion of Gene Expression Dataset with Annotated Dataset*: Genes determined to be co-expressed by clustering techniques are not necessarily co-regulated and hence may not have similar functions [53]. A possible approach may be that the annotated subset of differentially expressed genes clustered together based on functional similarity could be superimposed on top of co-expressed genes leading to stability of clustering results. One such approach to fuse the expression dataset with the annotated dataset and other related functional annotation information has been proposed in [54].

vi. *Similarity Measure and Clustering Solution*: One of the main issues in subspace clustering is the definition of similarity, taking into account only certain subspaces. Different subspaces may be derived by specifying different weights, different selections, or even different combinations of attributes of a data set, to which a desirable similarity model has been applied. Since the subspace is not necessarily the same for different clusters within one clustering solution, this selection of a “desirable” similarity model is a task that cannot be accomplished independent of the clustering solution. Hence subspace clustering algorithms cannot be thought of as traditional clustering algorithms using just a different definition of similarity, rather, the similarity measure and the clustering solution are dependent on each other and are to be derived simultaneously.

5. CONCLUSION

In this paper, we have made an attempt to identify the problems associated with clustering of gene expression data, using traditional clustering methods, mainly due to the high dimensionality of the data involved. For this reason, subspace clustering techniques can be used to uncover the complex relationships found in data since they evaluate features only on a subset of the data. Differentiating between the nearest and the farthest neighbors becomes extremely difficult in high dimensional data spaces. Hence a thoughtful choice of the proximity measure has to be made to ensure the effectiveness of a clustering technique. The automated acquisition of data in many application domains gives rise to collection of many redundant features, ultimately interfering with the identification of “true” clusters. Moreover, the substitution of missing values and handling of “noise” may introduce inaccuracies and inconsistencies, leading to erroneous conclusions about the data by the domain experts.

The authors feel that genes determined to be co-expressed based on clustering techniques may not have always similar functions and hence may not be co-regulated. Therefore, one can also explore the possibility of superimposing the clusters obtained with the annotated subset of differentially expressed genes clustered together based on functional similarity.

It is well known that most clustering methods are highly variable 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 Gene Ontology, could be incorporated using data fusion earlier in the analysis of genomic data, the additional information about genes and their relationship with each other may improve stability, accuracy and/or biological relevance of the clusters.

REFERENCES

[1] Han, J. and Kamber, M. 2001. Data Mining: Concepts and Techniques, chap. 8, 335–393. Morgan Kaufmann Publishers.

[2] Berkhin, P. 2002. Survey of Clustering Data Mining Techniques. http://citeseer.nj.nec.com/berkhin02survey.html.

[3] Xu, R. and Wunsch, D. 2005 Survey of Clustering Algorithms. IEEE Transactions on Neural Networks. 16(3), 645-678.

[4] Nagi, S., Bhattacharyya, D.K. and Kalita, J. 2011. Gene Expression Data Clustering: A Survey. In: Proceedings of the 2nd National Conference on Emerging Trends and Applications in Computer Science (NCETACS) DOI: 10.1109/NCETACS.2011.5751377.

[5] Ahmed, H., Mahanta, P., Bhattacharyya, D.K. and Kalita, J.K. 2011. Gerc: tree based clustering for gene expression data. In: 2011 IEEE 11th international conference on Bioinformatics and Bioengineering (BIBE), IEEE, New York. 299–302.

[6] Mahanta, P., Ahmed, H., Bhattacharyya, D.K. and Kalita, J. 2012. An effective method for network module extraction from microarray data. BMC Bioinformatics 2012. 13(Suppl 13):S4.

[7] Parson, L., Haque, E.and Liu, H. 2004. Subspace Clustering for High Dimensional Data: A Review. SIGKDD Explorations. 6(1), 90–105.

[8] Bellman, R. 1961. Adaptive Control Processes - A Guided Tour. Princeton University Press.

[9] Beyer, K., Goldstein, J., Ramakrishnan, R. and Shaft, U. 1991. When is “nearest neighbor” meaningful? In: 7th Int. Conf. on Database Theory.

[10] Hinneburg, A., Aggarwal, C.C. and Keim, D.A. 2000. What is the nearest neighbor in high dimensional spaces? In: 26th Int. Conf. on Very Large Data Bases (VLDB).

[11] Agrawal, R., Gehrke, J., Gunopulos, D. and Raghavan P. 1998. Automatic subspace clustering of high dimensional data for data mining applications. In: 1998 ACM SIGMOD international conference on Management of data. 94-105. ACM Press.

[12] Raychaudhuri, S., Sutphin, P.D., Chang, J.T. and Altman, R.B. 2001. Basic Microarray Analysis: Grouping and Feature Reduction. Trends in Biotechnology. 19(5), 189–193.

[13] Cha, S.-H. 2007. Comprehensive survey on distance/similarity measures between probability density functions. Int. J. Mathematical Models and Methods in Appl. Sc. 1(4).

[14] Cheng, C.H., Fu, A.W. and Zhang, Y. 1999. Entropy-based subspace clustering for mining numerical data. In: 5th ACM SIGKDD International Conference on Knowledge discovery and data mining. 84-93, ACM Press.

[15] Goil, S., Nagesh, H. and Choudhary, A. 1999. Mafia: Efficient and scalable subspace clustering for very large datasets. In: Technical Report CPDC-TR-9906-010, Northwestern University, 2145 Sheridan Road, Evanston IL 60208.

[16] Chang J.W. and Jin, D.S.: 2002. A new Cell-Based Clustering method for large, high-dimensional data in data mining applications. In: 2002 ACM symposium on Applied computing. 503-507, ACM Press.

[17] Liu, B., Xia, Y. and Yu, P.S. 2000. Clustering through decision tree construction. In: 9th Int. Conf. on Information and knowledge management. 20-29, ACM Press.

[18] Procopiuc, C.M., Jones, M., Agarwal, P.K. and Murali, T.M. 2002. A Monte Carlo algorithm for fast projective clustering. In: 2002 ACM SIGMOD International Conference on Management of data. 418-427, ACM Press.

[19] Aggarwal, C.C., Wolf, J.L., Yu, P.S., Procopiuc, C. and Park, J. S. 1999. Fast algorithms for projected clustering. In. 1999 ACM SIGMOD Int. Conf. on Management of data. 61-72, ACM Press.

[20] Aggarwal, C.C. and Yu, P.S. 2000. Finding generalized projected clusters in high dimensional spaces. In: 2000 ACM SIGMOD Int. Conf. on Management of data. 70-81, ACM Press.

[21] Woo K.-G. and Lee, J.-H. 2002. FINDIT: a Fast and Intelligent Subspace Clustering Algorithm using Dimension Voting: PhD thesis, Korea Advanced Institute of Science and Technology, Taejon, Korea.

[22] Yang, J., Wang, W., Wang H, and Yu, P. 2002. δ-clusters: Capturing Subspace Correlation in a large dataset. In: 18th Int. Conf. on Data Engineering. 517-528.

[23] Friedman, J.H. and Meulman, J.J. 2002 Clustering objects on subsets of attributes, http://citeseer.nj.nec.com/friedman02clustering.html.

[24] Cheng Y. and Church, G.M. 2000. Biclustering of Expression Data. In: 8th Int. Conf. Intelligent Systems for Mol. Biol. (ISMB ’00), 93-103.

[25] Madeira, S.C. and Oliveira A.L. 2004. Biclustering Algorithms for Biological Data Analysis: A Survey. IEEE/ACM Transactions on Computational Biology and Bioinformatics 1(1), 24-45.

[26] Yang, J., Wang, W., Wang, H. and Yu, P. 2003. Enhanced Biclustering on Expression Data. In: 3rd IEEE Conf. Bioinformatics and Bioeng. 321-327.

[27] Ben-Dor, A., Chor, B., Karp, R. and Yakhini, Z. 2002. Discovering local structure in gene expression data: The order–preserving submatrix problem. In: 6th Int. Conf. on Computational Biology (RECOMB’02). 49–57.

[28] Murali, T. M. and Kasif, S. 2003. Extracting conserved gene expression motifs from gene expression data. In: Pacific Symposium on Biocomputing. 8, 77–88.

[29] Hartigan, J. A. 1972. Direct clustering of a data matrix. J. of the American Statistical Association (JASA). 67(337), 123–129.

[30] Duffy, D. and Quiroz, A. 1991. A permutation based algorithm for block clustering. J. of Classification. 8, 65–91.

[31] Tibshirani, R., Hastie, T., Eisen, M., Ross, D., Botstein, D. and Brown, P. 1999. Clustering methods for the analysis of DNA microarray data. Technical report, Department of Health Research and Policy, Department of Genetics and Department of Biochemistry, Stanford University.

[32] Tanay, A., Sharan, R. and Shamir, R. 2002. Discovering statistically significant biclusters in gene expression data. Bioinformatics. 18(Suppl. 1), S136–S144.

[33] Getz, G., Levine, E. and Domany, E. 2000. Coupled Two-Way Clustering Analysis of Gene Microarray Data. In*: Natural Academy of Sciences US*. 12079-12084.

[34] Tang, C., Zhang, L., Zhang, I. and Ramanathan, M. 2001. Interrelated Two-Way Clustering: An Unsupervised Approach for Gene Expression Data Analysis. In: 2nd IEEE Int. Symp. Bioinformatics and Bioeng. 41-48.

[35] Busygin, S., Jacobsen, G. and Kramer, E. 2002. Double Conjugated Clustering Applied to Leukemia Microarray Data. In: 2nd SIAM Int. Conf. Data Mining, Workshop Clustering High Dimensional Data.

[36] Zaki, M.J. and Zhao, L. 2005. TriCluster: An Effective Algorithm for Mining Coherent Clusters in 3D Microarray Data. In: SIGMOD 2005.

[37] Jiang, H., Zhou, S., Guan, J. and Zheng, Y. 2006. gTRICLUSTER: A More General and Effective 3D Clustering Algorithm for Gene-Sample-Time Microarray Data. In Proc. BioDM. 48-59.

[38] Li, A. and Tuck, D. 2009. An Effective Tri-Clustering Algorithm Combining Expression Data with Gene Regulation Information Gene Regul Syst Bio. 3: 49–64.

[39] Mahanta, P., Ahmed, H., Bhattacharyya, D.K. and Kalita, J. 2011. Triclustering in Gene Expression Data Analysis: A Selected Survey. In: Proceedings of the 2nd National Conference on Emerging Trends and Applications in Computer Science (NCETACS) DOI: 10.1109/NCETACS.2011.5751409.

[40] Ahmed, H., Mahanta, P., Bhattacharyya, D.K. and Kalita, J.K. 2012. Module Extraction from Subspace Co-expression Networks. Network Modeling Analysis in Health Informatics and Bioinformatics. 1(4), 183-195.

[41] Bohm, C., Kailing, K., Kroger P. and Zimek, A. 2004. Computing Clusters of Correlation Connected objects. In: ACM SIGMOD Int. Conf. on Management of Data.

[42] Rand, W.M. 1971. Objective criteria for the evaluation of clustering methods. J. American Statistical Association. 66(336), 846–850.

[43] Sharan, R., Maron-Katz, A. and Shamir, R. 2003. CLICK and EXPANDER: A System for Clustering and Visualizing Gene Expression Data. Bioinformatics. 19(14), 1787–1799.

[44] Rousseeuw, P. 1987. Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. J. Computational and Applied Math. 20, 153–165.

[45] Gibbons, F. and Roth, F. 2002. Judging the Quality of Gene Expression Based Clustering Methods using Gene Annotation. Genome Research.12, 1574–1581.

[46] Tavazoie, S., Hughes, D., Campbell, M.J., Cho, R.J. and Church, G.M. 1999. Systematic Determination of Genetic Network Architecture. Nature Genetics. 281-285.

[47] Jiang, D., Pei, J. and Zhang, A. 2003. Interactive Exploration of Coherent Patterns in Time-Series Gene Expression Data. In: 9th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining.

[48] Jiang, D., Pei, J. and Zhang, A. 2003. DHC: A Density-Based Hierarchical Clustering Method for Time-Series Gene Expression Data. In: BIBE2003: 3rd IEEE Int. Symp. Bioinformatics and Bioeng.

[49] Aggarwal, C.C., Hinneburg, A. and Keim, D. 2001. On the surprising behavior of distance metrics in high dimensional space. In: 8th Int. Conf. on Database Theory (ICDT).

[50] Costa, I.G., de Carvalho, F.A. and de Souto, M.C. 2004. Comparative analysis of clustering methods for gene expression time course data. Genet. Mol. Biol. 27, 623-631.

[51] Bandyopadhyay, S. and Bhattacharyya, M. 2010. A Biologically Inspired Measure for Co-Expression Analysis. IEEE/ACM Trans. on Comp. Biol. and Bioinf.

[52] Irigoien, I., Vives, S. and Arenas, C. 2009. Microarray Time Course Experiments: Finding Profiles. IEEE/ACM Trans. on Comp. Biol. and Bioinf.

[53] Loganantharaj, R. 2009. Beyond clustering of array expressions. Int. J. Bioinformatics Res. Appl. 5(3), 329-348.

[54] Kustra, R. and Zagdanski, A. 2010. Data-Fusion in Clustering Microarray Data: Balancing Discovery and Interpretability. IEEE/ACM Trans. on Comp. Biol. and Bioinf. 7(1), 50-63.