ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 2, February 2015

Performance Analysis of Subspace Clustering Algorithms in Biological Data

Shilpi Chakraborty[1], Bijoyeta Roy[2]

Assistant Professor, B. P Poddar Institute of Management and Technology, Kolkata, West Bengal, India[1]

Assistant Professor, SMIT, Sikkim Manipal Institute of Technology, East Sikkim, India[2]

Abstract: Data clustering which is also called as Cluster Analysis is the unsupervised classification of data into various clusters. Clustering is a method of unsupervised learning which is generally implemented by various machine learning techniques. In this paper a specific comparison of three kinds of clustering is introduced and finally the cost function and loss function are calculated. For evaluating any clustering methods, calculation of the error percentage of the concerned method play an important factor. In this paper comparison is carried out on k-means clustering algorithms, hierarchical algorithms and density based algorithms. The main criteria where focus is given while comparing the clustering algorithms are: Scalability, Classes for dealing with noise and extra deposition, different dimensions of high levels etc.

Keywords: Clustering, K means, Hierarchical algorithms, Density based algorithms, Cluster.

I. INTRODUCTION

Clustering is the process of organizing data into meaningful easy accessible groups, and these groups are called clusters. This is not something new to computer science, but has existed since long back as classification and taxonomical areas. Clustering groups (clusters) objects according to their similarity in their characteristics, even though it is generally a field of unsupervised learning, knowledge about the type and source of the data has been found to be useful in selecting the better clustering algorithm. This type of clustering has generally found use in fields like content data mining. Clustering can be seen as a generalization of classification. In classification there is the knowledge about the object, the characteristics that are being looked for and the classifications available. So the aim is more similar to just finding "**Where to put the new object in**". Clustering on the other hand analyses the data and finds out the characteristics in it, either based on responses (supervised) or more generally without any responses (unsupervised).

Clustering can also be seen as the reduction in the number of bits required to convey information about a member, so that it can be said "*Go past the third tree and take a right*" instead of saying "*go past the large green thing with red berries then past the large green thing with thorns and then take a right*". It can be seen that there is much less information in the first definition but usually that is all required and not only is the extra information unnecessary it could also cause confusion. Hence it can be seen that clustering is a form of data abstraction. The most general definition is that given N items, it can be divided into k groups based on the measure of similarity between the items, such that items in a group can be called 'similar'. Here, it is assumed that N items are all points in an M-dimensional space, for the applications as defined below each of the axis's will be assigned a special meaning. It doesn't mean that clusters are just the set of points that are closest together. There can be clusters which are lines, curves or even complex shapes, and spirals. In actual applications each of the axis would be recording a different quantifiable data, e.g. : Here color clusters are used where X - Axis is the red color, Y-Axis the Blue color and Z-axis the Green Color, and hence the points that can be seen together represent close enough colors. The following Fig. 1 & 2 represents the color clusters.

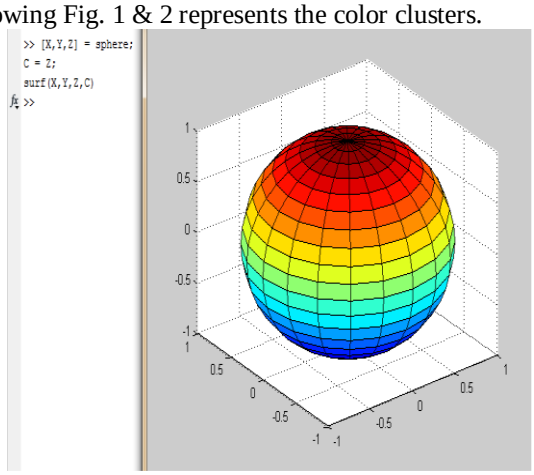

Figure1: Colored Clusters

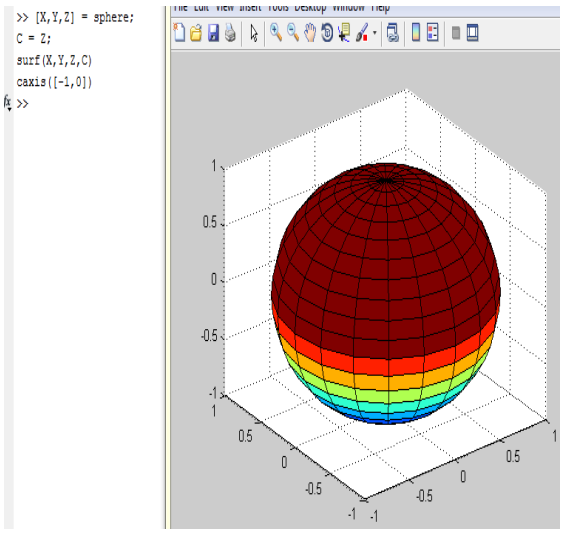

Figure 2: View of Colored Clusters

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 2, February 2015

Data analysis is of two types mainly (i) exploratory or descriptive which means that given a large set of data, main aim is to find the general characteristics and models present in the data and (ii) Confirmatory or inferential, where it will be given the characteristics for looking more into proving this item. It is seen that clustering serves as a formative process in the first model, where clustering itself will help in forming the high level model and the hypothesis, and in the second case even though not as strong as the first , however the inclusion of new elements can be done by checking if it would be included in the cluster or not Data clustering in 1,2 and 3 dimensions can be easily done by humans but as the number of dimensions increases there is more and more need of computers , and there exists problems that can be 8 – 10 dimensional.

In machine learning there are problems which exists in infinite dimensions and so on, where the only approach would be cluster analysis. The final note is that clustering just on the basis of physical closeness is not advisable. The figure from [Jain, 2010] given below shows the actual way clustering should happen, but if we go by physical closeness alone then will get erroneous results.

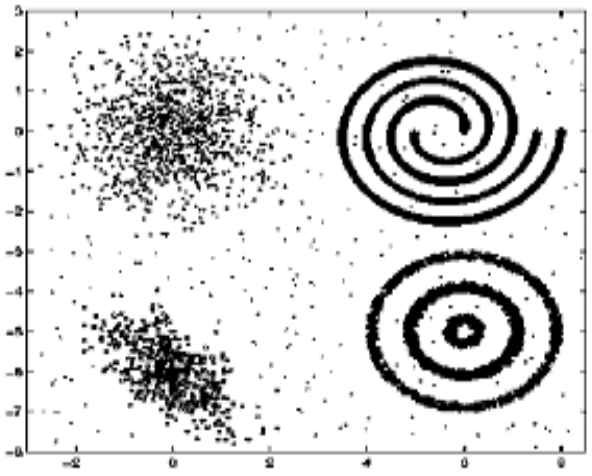

Figure 3: Input data

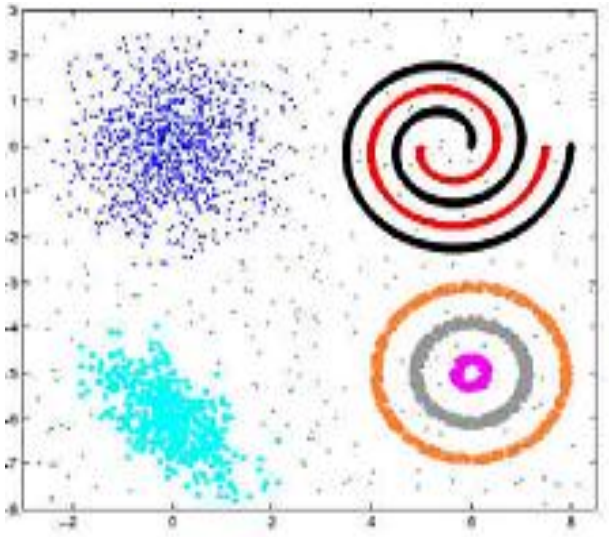

Figure 4: Desired Clustering

II. LITERATURE SURVEY:

Literature survey reveals various clustering techniques such as Hierarchical Clustering, K- Means Clustering, and Fuzzy based clustering etc.

A) Hierarchical Clustering: The operation of a hierarchical clustering algorithm is illustrated using the two-dimensional data set in Figure 5. This figure depicts seven patterns labeled A, B, C, D, E, F, and G in three clusters. A hierarchical algorithm yields a dendrogram representing the nested grouping of patterns and similarity levels at which groupings change. A dendrogram corresponding to the seven points in Figure 5 (obtained from the single-link algorithm [Jain and Dubes 1988]) is shown in Figure 6.

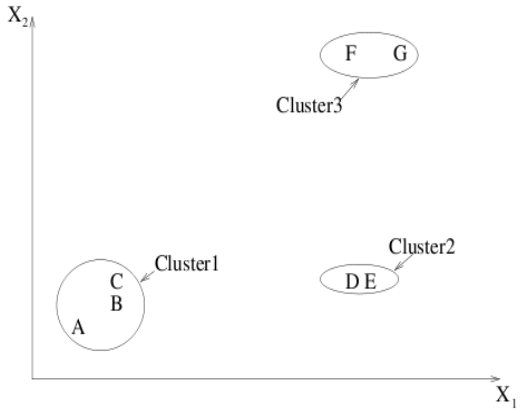

Fig 5: Hierarchical clustering

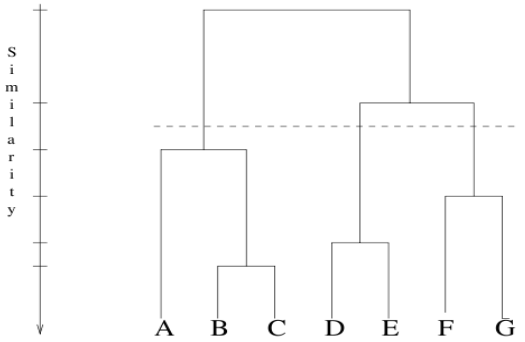

Fig 6: Dendogram of Hierarchical clustering

Most hierarchical clustering algorithms are variants of the single-link, complete-link, and minimum-variance algorithms. Of these, the single-link and complete-link algorithms are most popular. These two algorithms differ in the way they characterize the similarity between a pair of clusters. In the single-link method, the distance between two clusters is the minimum of the distances between all pairs of patterns drawn from the two clusters (one pattern from the first cluster, the other from the second). In the complete-link algorithm, the distance between two clusters is the maximum of all pair wise distances between patterns in the two clusters. In either case, two clusters are merged to form a larger cluster based on minimum distance criteria. The complete-link algorithm produces tightly bound or compact clusters. The single-link algorithm, by contrast, suffers from a chaining effect [Nagy 1968]. It has a tendency to produce clusters that are straggly or elongated. There are two clusters in Figures 7 and 8

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 2, February 2015

separated by a bridge of noisy patterns. The single-link algorithm produces the clusters shown in Figure 7, whereas the complete-link algorithm obtains the clustering shown in Figure 8. The clusters obtained by the complete link algorithm are more compact than those obtained by the single-link algorithm and the cluster labeled 1 obtained using the single-link algorithm is elongated because of the noisy patterns labeled. The single-link algorithm is more versatile than the complete-link algorithm, otherwise. However, from a pragmatic viewpoint, it has been observed that the complete link algorithm produces more useful hierarchies in many applications than the single-link algorithm.

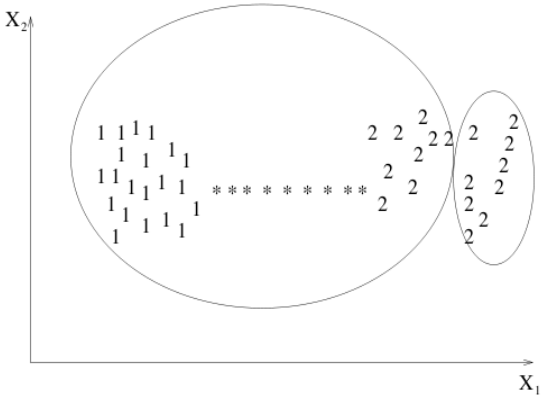

Figure 7: The single link Clustering

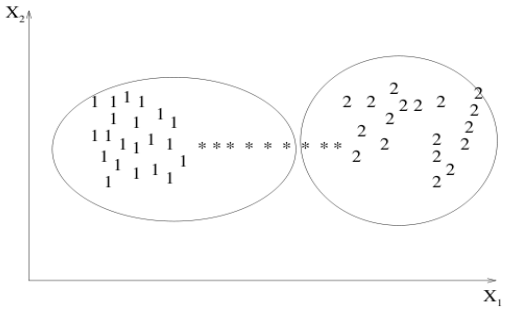

Figure 8: The Complete Link Clustering

i)Agglomerative Single Link Clustering Algorithm:

a) Place each pattern in its own cluster. Construct a list of inter pattern distances for all distinct unordered pairs of patterns, and sort this list in ascending order.

b) Step through the sorted list of distances, forming for each distinct dissimilarity value D_k a graph on the patterns where pairs of patterns closer than D_k are connected by a graph edge. If all the patterns are members of a connected graph, stop. Otherwise, repeat this step.

c) The output of the algorithm is a nested hierarchy of graphs which can be cut at a desired dissimilarity level forming a partition (clustering) identified by simply connected components in the corresponding graph.

ii) Agglomerative Complete Link Clustering Algorithm:

a) Place each pattern in its own cluster. Construct a list of inter pattern distances for all distinct unordered pairs of patterns, and sort this list in ascending order.

b) Step through the sorted list of distances, forming for each distinct dissimilarity value D_k a graph on the patterns where pairs of patterns closer than D_k are connected by a graph edge. If all the patterns are members of a completely connected graph, stop.

c) The output of the algorithm is a nested hierarchy of graphs which can be cut at a desired dissimilarity level forming a partition (clustering) identified by completely connected components in the corresponding graph. Hierarchical algorithms are more versatile than partitional algorithms. For example, the single-link clustering algorithm works well on data sets containing non-isotropic clusters including well-separated, chain-like, and concentric clusters, whereas a typical partitional algorithm such as the k –means algorithm works well only on data sets having isotropic clusters. On the other hand, the time and space complexities of the partitional algorithms are typically lower than those of the hierarchical algorithms. It is possible to develop hybrid algorithms that exploit the good features of both categories.

iii) Hierarchical Agglomerative Clustering Algorithm:

a) Compute the proximity matrix containing the distance between each pair of patterns. Treat each pattern as a cluster.

b) Find the most similar pair of clusters using the proximity matrix. Merge these two clusters into one cluster. Update the proximity matrix to reflect this merge operation.

c) If all patterns are in one cluster, then stop. Otherwise, go to step b. Based on the way the proximity matrix is updated in step b, a variety of agglomerative algorithms can be designed. Hierarchical divisive algorithms start with a single cluster of all the given objects and keep splitting the clusters based on some criterion to obtain a partition of singleton clusters.

B) K -Means Clustering Algorithm

a) Choose k cluster centers to coincide with k randomly chosen patterns or k randomly defined points inside the hyper volume containing the pattern set.

b) Assign each pattern to the closest cluster center.

c) Recompute the cluster centers using the current cluster memberships.

d) If a convergence criterion is not met, go to step b.

Typical convergence criteria are: no (or minimal) reassignment of patterns to new cluster centers, or minimal decrease in squared error. Several variants of the k -means algorithm have been reported in the literature. Some of them attempt to select a good initial partition so that the algorithm is more likely to find the global minimum value. Another variation is to permit splitting and merging of the resulting clusters. Typically, a cluster is split when its variance is above a pre-specified threshold, and two clusters are merged when the distance between their centroids is below another pre-specified threshold. Using this variant, it is possible to obtain the optimal partition starting from any arbitrary initial partition, provided proper threshold values are specified. The well-known ISO-DATA algorithm employs this technique of merging and splitting clusters.

The best-known graph-theoretic divisive clustering algorithm is based on construction of the minimal

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 2, February 2015

spanning tree (MST) of the data, and then deleting the MST edges with the largest lengths to generate clusters.

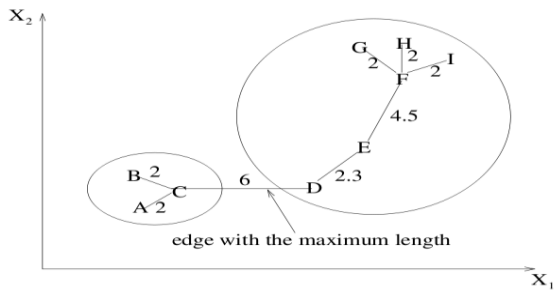

Figure 9: Clustering using Minimum Spanning Tree

Figure 9 depicts the MST obtained from nine two dimensional points. By breaking the link labeled CD with a length of 6 units (the edge with the maximum Euclidean length), two clusters (A, B, C and D, E, F, G, H, I) are obtained. The second cluster can be further divided into two clusters by breaking the edge EF, which has a length of 4.5 units. The hierarchical approaches are also related to graph-theoretic clustering. Single-link clusters are sub graphs of the minimum spanning tree of the data which are also the connected components. Complete-link clusters are maximal complete sub graphs, and are related to the node colorability of graphs. The maximal complete sub graph was considered the strictest definition of a cluster.

C) Fuzzy Clustering

Traditional clustering approaches generate partitions and in a partition, each pattern belongs to one and only one cluster. Hence, the clusters in a hard clustering are disjoint. Fuzzy clustering extends this notion to associate each pattern with every cluster using a membership function. The output of such algorithms is a clustering, but not a partition. Following is a high level partitioned fuzzy clustering algorithm.

1. Initialize $U=[u_{ij}]$ matrix, $U^{(0)}$
2. At k-step: calculate the centers vectors $C^{(k)}=[c_j]$ with $U^{(k)}$

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

3. Update $U^{(k)}$, $U^{(k+1)}$

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}$$

4.If $\| U^{(k+1)} - U^{(k)} \| < \varepsilon$ then STOP; otherwise return to step 2.

In fuzzy clustering, each cluster is a fuzzy set of all the patterns. Here, the simple case of a mono-dimensional application of the FCM is considered. Twenty data and three clusters are used to initialize the algorithm and to compute the U matrix. Figures 10, 11 and 12 show the membership value for each datum and for each cluster. The color of the data is that of the nearest cluster according to the membership function.

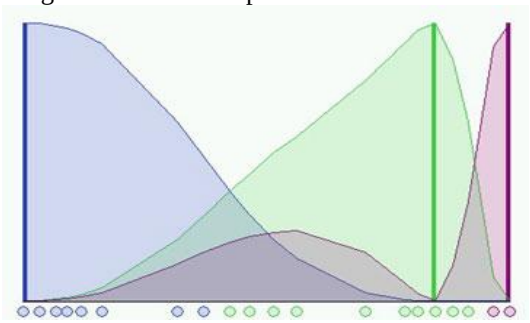

Fig 10: Initial condition where the fuzzy distribution depends on the particular position of the clusters

In the simulation shown in the figure above a fuzzyness coefficient m = 2 is used and the algorithm is also imposed to terminate when $\max_{ij} \left\{ \left| u_{ij}^{(k+1)} - u_{ij}^{(k)} \right| \right\} < 0.3$. The picture shows the initial condition where the fuzzy distribution depends on the particular position of the clusters. No step is performed yet so that clusters are not identified very well. Now the algorithm is ran until the stop condition is verified. The figure below shows the final condition reached at the 8th step with m=2 and ε =0.3:

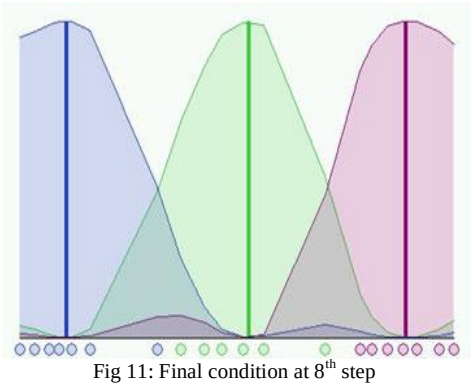

Fig 11: Final condition at 8th step

In the next figure a better result is seen having used the same initial conditions and ε=0.01, but 37 steps were required.

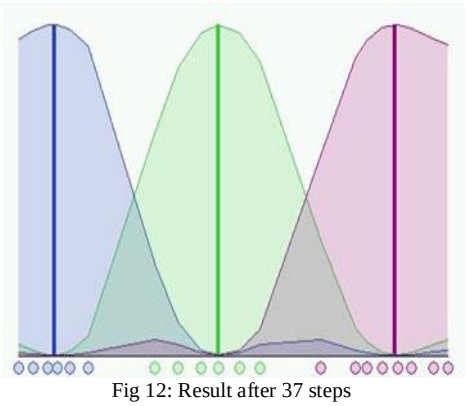

Fig 12: Result after 37 steps

It is also important to notice that different initializations cause different evolutions of the algorithm. In fact it could converge to the same result but probably with a different number of iteration steps.

DOI 10.17148/IJARCCE.2015.4259

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 2, February 2015

III. PROPOSED WORK:

[Jain and Dubes, 1988] gives a nice outline for the various steps involved in any clustering algorithm. In general the clustering algorithm can be seen as a pipe model with feedback, that is the output of each phase is send as input for the next phase, and if and when required the more clustered data is send back, so that new patterns can be analyzed. The various steps involved according to [Jain and Dubes, 1988] are:

1) Pattern representation - This is the first step and defines what the pattern is, it can also include feature extraction and selection. The inclusion depends on the type of application and the data we know about the distribution. Pattern representation refers to the number of classes, the number of available patterns, and the number, type, and scale of the features available to the clustering algorithm. Some of this information may not be controllable by the Data Clustering practitioner. Feature selection is the process of identifying the most effective subset of the original features to use in clustering. Feature extraction is the use of one or more transformations of the input features to produce new salient features. Either or both of these techniques can be used to obtain an appropriate set of features to use in clustering.

2) Definition of Pattern Proximity - This is defined appropriate to the data domain, and this is what will define how close the values are and whether they need to be clustered separately or together, for e.g. all patterns along a line might be clustered together and a spiral cluster could be made of many simple curve. This takes two patterns as the function and considers the output of the distance function between them and there can be a variety of distance function defined like

i) The Euclidean distance.
ii) The Manhattan distance
iii) The maximum norm (or infinity norm)
iv) The Mahalanobis distance.
v) The angle between two vectors (Generally used for higher dimensional data)
vi)The Hamming distance (Usually used to see errors and text analysis)

In addition to this specific distance functions can also be defined based on our input.

3) Clustering/Grouping - This is where the clustering is finally done this takes input of the patterns and the definition of pattern proximity based on these patterns and produces the output as distinct clusters. Even now the data is just a set of points and clusters, for some uses this will be the last step. The theoretical process of clustering ends here, but most models go on to define abstraction and finally get information out of them.

4) Data Abstraction - This is an optional step, but is usually done in almost all the algorithms. Here, simplicity is either from the perspective of automatic analysis (so that a machine can perform further processing efficiently) or it is human-oriented (so that the representation obtained is easy to comprehend and intuitively appealing). In the clustering context, a typical data abstraction is a compact description of each cluster, usually in terms of cluster prototypes or representative patterns such as the centroid [Diday and Simon 1976]. Hence, here the cluster is now defined in terms of a representational data, and unwanted individual characteristics of the points are thrown away, this will cause just the wanted information to remain, and hence less storage space and complexity.

5) Assessment - In few cases there is the study of cluster tendency, wherein the input data are examined to see if there is any merit to a cluster analysis prior to one being performed. Generally but this has not found much use . Cluster validity is a post clustering analysis where assessments are objective, so that it can be automated and allows no ambiguity are performed to determine the input data are examined to see if there is any merit to a cluster analysis is prior to one being performed. Generally but this has not found much use. Cluster validity is a post clustering analysis where assessments are objective, so that it can be automated and allows no ambiguity are performed to determine whether the output is meaningful. This is usually done in Machine Learning or similar situation where the clusters are tested for their accuracy by various means such as Getting Feedback from the deployment (Search Results), or processing with a different image file in image processing. The validation could be external -comparison to a known structure, internal – analyzing the structure based on the required properties, and checking inherent appropriateness. Relative - where two clustering's from different algorithms are checked and compared.

IV. EXPERIMENTAL RESULT

One of the sophisticated factor for clustering algorithms and evaluated is the average of error rate for each clustering method. But one of the important factors for calculating the error rate is that the result of the labeling of each clustering method maybe will be completely different in the test labels; for example, K value is equal to 2 and clustering method finds two different labels, so the error rate of this algorithm can be more than 50%. In the result of a data set, the result of data set was 99.98 % for the K is equal to 2. For clustering into two clusters, the error rate cannot be more than 50%, so the real data set of this data set is equal to 0.02% because all of the clustering algorithm gets the labels, and cannot find exact labels, so the real error rate is equal to minimum of the error rate by different labels of K.

For K = 2:

Error Rate = min {|L – TestLabels |, 100 - |L – TestLabels |}

For K > 2

Loop 1 to 2k

Error Rate = min {|L – Test Labels |}

This method is only available for k = 2, so a general algorithm is required to find error rate by these two labels for all value of K, which can calculate the error rate. For example, if there are 3 different clusters, clusters cannot be found by one comparison. This method has exponential time complexity, but by the value of K not by N and N>>K which K is very smaller than N data points; for example, for the first data set clustering is done on 14977 data points.

DOI 10.17148/IJARCCE.2015.4259

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 2, February 2015

In Data set 1 with 14977 *2 dimensions by k=2 SP clustering has 17 error so the error rate is 0.11% but with K-means which has 2902, error rate is 19.37%.

TABLE I: Data Set of 2000 Data points (Genes) and with 15 samples (dimensions)

K	2	3	5	10
K-means	19.37%	30.2 %	40%	56%
Hierarchical Clustering	43.2%	49%	51%	53.6%

TABLE II: Two moon Data Sets 14977 data points with two dimensions

K	2	3	5	10
K-means	49%	43%	59%	86%
Hierarchical Clustering	59%	39%	49%	79%

***Results of Fuzzy C means algorithm*:**

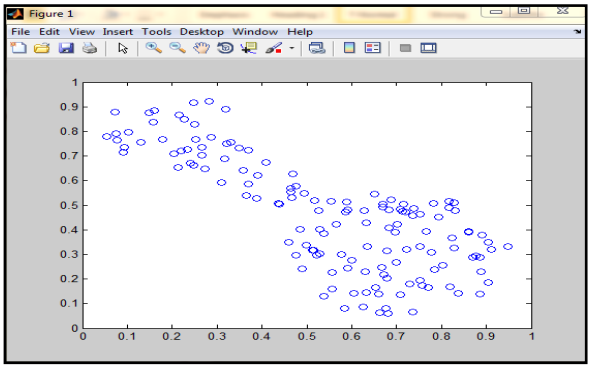

Figure 13: Data Set

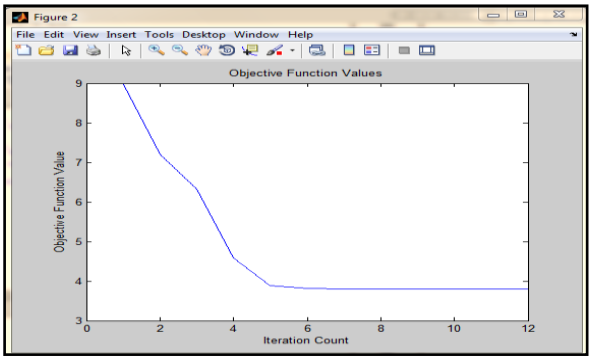

Figure 14: Objective Function value

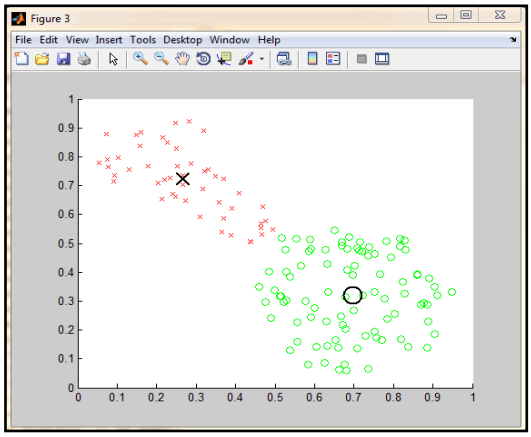

Figure15: Clustered data

V. CONCLUSION

In this paper, existing projected and subspace clustering literature is discussed. A comparative study is carried out between three available algorithms, and their advantages and disadvantages are pointed out. In general, Hierarchical clustering algorithm is better in terms of time for calculation and it obtained the least number of unclustered data. K means is better for simple techniques whereas if accuracy is considered hierarchical is relevant for various attributes found.

REFERENCES

[1] C. S. Li, "Cluster Center Initialization Method for K-means Algorithm Over Data Sets with Two Clusters", International Conference on Advances in Engineering, Elsevier, vol.24, pp. 324-328, 2011.

[2] M. Erisoglu, N. Calis and S. Sakallioglu, "A new algorithm for initial cluster centers in K-Means algorithm", Pattern Recognition Letters, vol. 32, issue 14, Oct.2011.

[3] N. Mehta S. Dang "A Review of Clustering Techniques in various Applications for effective data mining" International Journal of Research in Engineering & Applied Science vol. 1, No. 1 2011.

[4] S.Thiprungsri, M. A. Vasarhelyi."Cluster Analysis for Anomaly Detection in Accounting Data: An Audit Approach" The International Journal of Digital Accounting Research Vol. 11, 2011, pp. 69 - 84

[5] B. Rama et. Al., "A Survey on clustering Current Status and challenging issues"(IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 9, pp. 2976-2980, 2010.

[6] O. A Abbas," Comparisions between Data Clustering Algorithms", The International Arab Journal of Information Technology, Vol. 5, No. 3, 2008.

[7] P. Baser,J. R. Saini," A Comparative Analysis of various Clustering Techniques used for very Large Datasets",International Journal of Computer Science and Communication Networks, Vol. 3, No. 4, 2008, pp 271-275.

[8] J. C. Dunn (1973): "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters", *Journal of Cybernetics* 3: 32-57

[9] J. C. Bezdek (1981): "Pattern Recognition with Fuzzy Objective Function Algoritms", Plenum Press, New York