International Journal of Engineering Science Invention (IJESI)
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org ||Volume 8 Issue 03 Series. I || March 2019 || PP 52-64

A Survey on Semi Supervised Clustering For High Dimensional Data Clustering

M. Pavithra[1],Dr.R.M.S.Parvathi [2.]
Assistant Professor, Department of C.S.E, Jansons Institute of Technology, Coimbatore, India[1].
Dean- PG Studies, Sri Ramakrishna Institute of Technology, Coimbatore, India[2].
Corresponding Author: M. Pavithra

Abstract: *Cluster formation has three types as supervised clustering, unsupervised clustering and semi supervised. Clustering algorithms are based on active learning, with ensemble clustering-means algorithm, data streams with flock, fuzzy clustering for shape annotations, Incremental semi supervised clustering, Weakly supervised clustering, with minimum labeled data, self-organizing based on neural networks. Semi-supervised clustering is combination of supervised clustering and unsupervised clustering. It has an important impact on clustering [2]. Clustering ensemble is one of the most recentadvances in unsupervised learning. It aims to combine theclustering results obtained using different algorithms or fromdifferent runs of the same clustering algorithm for the samedata set, this is accomplished using on a consensus function, theefficiency and accuracy of this method has been proven in manyworks in literature.It introduces a method of clustering based on pair-wise constraints [3]. This method uses neighborhood framework and select most informative point. By performing the query against all data points, data points are clustered. Therefore, a number of semi-supervised clustering algorithms have been proposed, but few of them are specially designed for high dimensional data. High dimensionality is a difficult challenge for clustering analysis due to the inherent sparsedistribution, and most of popular clustering algorithms including semi-supervised ones will be invalid in high dimensional space. A semi-supervised hierarchical clusteringalgorithm for high dimensional data is proposed, which is based on the combination of semisupervised clustering and dimensionality reduction [1]. In order to achieve high harmony betweendimensionality reduction and inherent cluster structure detection, the number of dimensions isreduced sequentially as the clusters are gradually formed in the hierarchical clusteringprocedure.Finding clusters in high dimensional data is a challenging task asthe high dimensional data comprises hundreds of attributes [4].Subspace clustering is an evolving methodology which, insteadof finding clusters in the entire feature space, it aims at findingclusters in various overlapping or non-overlapping subspaces ofthe high dimensional dataset. Density based subspace clusteringalgorithms treat clusters as the dense regions compared to noiseor border regions. Many momentous density based subspaceclustering algorithms exist in the literature [5]. Each of them ischaracterized by different characteristics caused by differentassumptions, input parameters or by the use of differenttechniques etc. Hence it is quite unfeasible for the futuredevelopers to compare all these algorithms using one commonscale [6].The aim of Semi-supervised clustering algorithm is to improve theclustering performance by considering the user supervision based onthe pairwise constraints. In this paper, we examine the active learningchallenges to choose the pairwise must-link and cannot-linkconstraints for semi-supervised clustering [7].The process of grouping into high dimensional data into clusters is not accurate and perhaps not up to the level of expectation when the dimension of the dataset is high. It is now focusing tremendous attention towards research anddevelopment [8]. The performance issues of the data clustering in high dimensional data it is necessary to study issues likedimensionality reduction, redundancy elimination, subspace clustering, co-clustering and data labeling for clusters are to analyzedand improved [9].*

Date of Submission: 17-02-2019 Date of acceptance: 03-03-2019

I. Introduction

Clustering is method of creating number of clusters (groups) of large amount of data. The data points that lie in same cluster are similar to each other in terms of their features and the data points that lie in different clusters are dissimilar to each other in terms of their features. So, it is advantageous to have similar data together [10]. Three types of clustering methods are there: Unsupervised, Supervised, Semi-supervised. In unsupervised clustering,the features of all the data points are already known; according to those features clustering is performed [11]. So it is one of simple method of clustering. In case of supervised clustering, the features of data points are not known; the features need to be extracted before performing clustering. Semi-supervised clustering is the combination of unsupervised clustering and supervised clustering [12]. It employs both labeled and unlabeled data. The features of some data points are known but not of all data points [13].

The huge and amount of datathat is generated by this communication processcontains important information that accumulates daily indatabases and is not easy to extract. The field of datamining developed as a means of extracting informationand knowledge from databases to discover patterns or concepts that are not evident. So, it is esteemed thatthere is a mounting need for a more sophisticated automated system of partitioning the datasets intogroups, or clusters [14]. Clustering is defined as the process of finding a structure where the data objects aregrouped into clusters which are similar behavior". Forexample, as digital libraries and the World Wide Webare growing exponentially, the ability to find usefulinformation progressively depends on the indexinginfrastructure or search engine. Clustering techniquescan be used to discover natural groups in data sets and toidentify a structure that might reside there, withouthaving any specific background knowledge ascharacteristics of the data [15].

Clustering can be considered as the most importantunsupervised learning problem. Clustering deals with finding a primitive structure in a collection of unlabeleddata [16]. A cluster a collection of objects which are "similar" between them and are "dissimilar" to theobjects belonging to other clusters [2]. The objective ofthe clustering technique is to determine the intrinsicgrouping in a set of unlabeled data. The similarity between data objects can be measured with the imposeddistance values [17]. Specifying the distance measures forthe high dimensional data is becoming very trivialbecause it holds different data values in their corresponding attributes. Following is the analysis ofdifferent distance measures used for measuring similarity between data objects in clustering [3].

Some semi-supervised clustering algorithms have been presented [4], but few ofthem are specially designed for high dimensional data. In most of clusteringapplications such as image processing, pattern recognition, computational biology,and web information retrieval, the data need to be processed are always in high dimensional space [5]. High dimensionality not only makes computational cost very expensive, but also makes many popular clustering algorithms invalid due to sparsedensity distribution. Therefore, the curse of dimensionality must be given a significantamount of research attention in semi-supervised clustering [6].Dimensionality reduction is thought as an effective way to solve high dimensional problem. In most cases, dimensionality reduction is carried out as a preprocessingstep, for example, linear/nonlinear discriminant analysis (LDA/NDA) and principalcomponent analysis (PCA) are popular used in classification and clustering problemsrespectively [7].

Therefore, most of studies of high-dimensional data clustering always use more complicated schemes to incorporate dimensionality reduction into clustering procedure instead of using dimensionality reduction as preprocessing step,i.e., two problems of partitioning a data set and finding reduced dimensionalities are solved at the same time [8]. Semi-supervised clustering is another hot topic in machinelearning, in which both labeled and unlabeled data are usedfor training - typically a small amount of labeled data with a large amount of unlabeled data [9]. Semi-supervised learningfalls between supervised learning (with completely labeledtraining data) and unsupervised learning (without anylabeled training data). It has been proven by many machinelearning researchers that unlabeled data, when used inconjunction with a small amount of labeled data, canproduce considerable improvement in learning accuracy [10]. Generating labeled data for a learning problem oftenrequires a human effort (supervision) to manually classifytraining examples [11]. The cost producing a fully labeledtraining set in the labeling process may be infeasible in manycases, whereas producing unlabeled data is relativelyinexpensive. In such situations, semi-supervised learning canbe of great practical value [12].

Many semi-supervised algorithms were proposed inliterature with various methodologies, some based on EMwith generative mixture models, self-training, co-training,Transductive Support Vector Machines (TSVM), and graphbased methods [13]. Because labeled data is scarce, semisupervised learning methods make strong modelassumptions. Ideally we should use a method whoseassumptions fit the problem structure [14]. This may be difficultin reality. Generally, EM with generative mixture modelsmay be a good choice if the classes produce well clustereddata; co-training may be appropriate when the features naturally split into two sets; graph-based methods can beused if points with similar features tend to be in the sameclass. But there is no direct way for choosing the type ofsemi-supervised algorithm [15].

II. Related Work

In the last few years many research works have been done on high-dimensional data clustering and evolving data streams clustering. There are extensive research works on clustering algorithms for static datasets [5], [6], [4] where some of them have been further extended for evolving data streams. The clusters are formed based on a Euclidean distance function like k-means algorithm [7]. K-mean clustering splits the nd dimensional points into k cluster ($k < n$). One of the well-known extensions of k-means on data streams is presented [8]. They propose an algorithm called CluStream based on k-means for clustering evolving data streams. CluStream introduces an online-offline method for clustering data streams. CluStream clustering idea is adopted in the majority of data stream clustering algorithms. It extended their work in HPStream [9], which introduces the projected clustering to data streams [2]. In projected clustering high dimensional stream data is

partitioned based on the preferred dimensions instead of full the dimensional space [1]. It uses the density-based clustering without projected dimensions in DenStream algorithm. For streaming data, although a considerable research has tackled the fullspace clustering, relatively limited work deals with the subspace clustering. These few researches include [9] HPStream, [11] HDDStream, and [12] SubCMM. A more comprehensive review and classifications are given in survey [13].

The active learning framework is used for document clustering. This framework uses an iterative approach. Here, foreach pair of documents, the probability of them belonging to the same cluster is computed and measures the associated uncertainty [4]. By checking the pair-wise constraints it performs clustering. There exist few excellent surveys on high dimensional data clustering approaches in literature [3]. In [2], authors have presented a variety of algorithms and challenges for clustering gene expression data. They also discussed different methods of cluster validation and cluster quality assessment. The authors in [12] have presented an extremely comprehensive survey beginning with the illustration of different terminologies used in subspace clustering methodologies. It discusses various assumptions, heuristics or intuitions forming the basis of different high dimensional data clustering approaches. In [10], authors have explored the behavior of some of the grid based subspace clustering algorithms. However, there does not exist any explicit comparison among all existing density based subspace clustering algorithms [5]. We present in this paper, such a comparison among clustering algorithms which adopts density based subspace clustering approach for clustering high dimensional data [6].
The novel scheme exploits both semi-kernel learning and batch mode active learning for relevance feedback in CBIR. In particular, a kernel function is first learned from a mixture of labeled and unlabeled examples [7]. The kernel will then be used to effectively identify the informative and diverse examples for active learning via a min-max framework [13]. An empirical study with relevance feedback of CBIR showed that the proposed scheme is significantly more effective than other state-of-the-art approaches [8].

Learning with user's interactions is crucial to many applications in computer vision and pattern recognition. One ofthem is content-based image retrieval (CBIR) where users are often engaged to interact with the CBIR system for improving the retrieval quality [9]. Such an interactive procedure is often known as relevance feedback, where the CBIR system attempts to understand the user's information needs by learning from the feedback examples judged by users [10]. Due to the challenge of the semantic gap, traditional relevance feedback techniques often have to repeat many runs in order to achieve desirable results [11]. To reduce the number of labeled examples required by relevancefeedback, one key issue is how to identify the most informative unlabeled examples such that the retrieval performance could be improved most efficiently [12]. Active learning is an important technique to address this challenge.In particular, we presented a unified learning framework for incorporating both labeled and unlabeled data to improve the retrieval accuracy, and developed a new batch mode active learning algorithm based on the min-max framework [13]. The empirical results with relevance feedback of CBIR showed the advantages of the proposed solution compared to the other state of-the-art methods.

In the existing system, clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar to each other than to those in other groups (clusters). More relevant to our work is an active learning framework presented by Huang and Lam [12] for the task of document clustering. Specifically, this framework takes an iterative approach that is similar to ours. In each iteration, their method performs semi-supervised clustering [1], [5], [13], [14]with the current set of constraints to produce a probabilisticclustering assignment.It then computes, for each pair ofdocuments, the probability of them belonging to the same clusterand measures the associated uncertainty [2]. To make a selection, itfocuses on all unconstrained pairs that has exactly one documentalready "assigned to" one of the existing neighborhoods by thecurrent constraint set, and among them identifies the mostuncertain pair to the query [3]. If a "must-link" answer is returned, itstops and moves onto the next iteration. Otherwise, it will querythe unassigned point against the existing neighborhoods until a"must-link" is returned [4].

Finally, we want to mention another line of work thatuses active learning to facilitate clustering [7], [8], where thegoal is to cluster a set of objects by actively querying thedistances between one or more pairs of points. This is differentfrom the focus of this paper, where we only request pairwisemust-link and cannot-link constraints, and do not require the userto provide specific distance values.In the existing system, Constraints and Table labels are notused [5]. In previous Must link, cannot link constraints are not usedand algorithms are not used its be a major drawback. Anabnormal clustering result will lead to meaningless models andpoor variety of queries [6].

III. Semi-Supervised Clustering

Semi-supervised clustering, which uses class labels or pairwise constraints on some examples to aid unsupervisedclustering. If the initial labeled data represent all the relevant categories, then both semi-supervised clustering and semi-supervised classification algorithms can be used for categorization [7]. However in many domains, knowledge of the relevant categories is incomplete. Unlike semi-supervised classification, semi-

supervised clustering (in the model-selection framework) can group data using the categories in the initial labeled data as well as extend and modify the existing set of categories as needed to reflect other regularities in the data [8].

III.A.Semi-Supervised Clustering Using Labeled Data

In this section, we give an outline of our initial work where we considered a scenario where supervision is incorporated into clustering in the form of labeled data [9]. We used the labeled data to generate seed clusters that initialize a clustering algorithm, and used constraints generated from the labeled data to guide the clustering process [10]. The underlying intuition is that proper seeding biases clustering towards a good region of the search space, thereby reducing the chances of it getting stuck in poor local optima, while simultaneously producing a clustering similar to the user specified labels. The importance of good seeding in clustering is well-known [11]. In partitional clustering algorithms like EM or K-Means, some commonly used approaches for initialization include simple random selection, taking the mean of the whole data and randomly perturbing to get initial cluster centers, or running K smaller clustering problems recursively to initialize [12]. Some other interesting initialization methods include the Buckshot method of doing hierarchical clustering on a sample of the data to get an initial set of cluster centers, running repeated K-Means on multiple data samples and clustering the K-Means solutions to get initial seeds, and selecting the K densest intervals along each co-ordinate to get the K cluster centers [13]. Our approach is different from these because we use labeled data to get good initialization for clustering.

III.A1.Problem Definition

Given a dataset, as previously mentioned, KMeans clustering of the dataset generates a -partitioning of so that the KMeans objective is locally minimized. Let, called the seed set, be the subset of data-points on which supervision is provided as follows: for each, the user provides the cluster of the partition to which it belongs [14]. We assume that corresponding to each partition of, there is typically at least one seed point. Note that we get a disjoint -partitioning of the seed set, so that all belongs to according to the supervision. This partitioning of the seed set forms the seed clustering. The goal is to guide the KMeans algorithm towards the desired clustering of the whole data as illustrated by the seed clustering [15].

III.A2.Motivation Of Semi-Supervised Kmeans Algorithms

The two semi-supervised KMeans algorithms presented in the last section can be motivated by considering KMeans in the EM framework, as shown. The only "missing data" for the KMeans problem are the conditional distributions of the cluster labels given the points and the parameters, Knowledge of these distributions solves the clustering problem, but normally there is no way to compute it [16]. In the semi-supervised clustering framework, the user provides information about some of the data points that specifies the corresponding conditional distributions [17]. Thus, semi-supervision by providing labeled data is equivalent to providing information about the conditional distributions. In standard KMeans without any initial supervision, the means are chosen randomly in the initial M-step and the data-points are assigned to the nearest means in the subsequent E-step [2]. As explained above, every point in the dataset has possible conditional distributions associated with it corresponding to the means to which it can belong [1]. This assignment of data point to a random cluster in the first E-step is similar to picking one conditional distribution at random from the possible conditional distributions [3].

III.B.Semi-Supervised Clustering Using Pairwise Constraints

In this work, we considered a framework that has pairwise must-link and cannot-link constraints between points in a dataset (with an associated cost of violating each constraint), in addition to having distances between the points [4]. These constraints specify that two examples must be in the same cluster (must-link) or different clusters (cannot-link). In real-world unsupervised learning tasks, e.g., clustering for speaker identification in a conversation, visual correspondence in multi-view image processing, clustering multi-spectral information from Mars images, etc., considering supervision in the form of constraints is generally more practical than providing class labels, since true labels may be unknown a priori,while it can be easier for a user to specify whether pairs of points belong to the same cluster or different clusters [5]. Constraints are a more general way to provide supervision in clustering than labels — given a set of labeled points one can always infer an unique equivalent set of pairwise must-link and cannot-link constraints, but not vice versa [6].We proposed a cost function for pairwise constrained clustering (PCC) that can be shown to be the energy of a configuration of a Markov random field (MRF) over the data with a well-defined potential function and noise model [7].

Then, the pairwise-constrained clustering problem becomes equivalent to finding the MRF configuration with the highest probability, or, in other words, minimizing its energy. We developed an iterative KMeans-type algorithm for solving this problem [8]. Previous work in the PCC framework includes the hard-

constrained COP-KMeans algorithm and the soft-constrained SCOP-KMeans algorithm, which have heuristically motivated objective functions [9]. Our formulation, on the other hand, has a well-defined underlying generative model. It also proposed a theoretical model for pairwise constrained clustering, but their clustering model uses only pairwise constraints for clustering, whereas our formulation uses both constraints and an underlying distance metric [10]. Pairwise clustering models have also been proposed for other non-parametric clustering algorithms [11].

III.C.Active Learning For Semi-Supervised Clustering

In order to maximize the utility of the limited supervised data available in a semi-supervised setting, supervised training examples should be actively selected as maximally informative ones rather than chosen at random, if possible [12]. In the PCC framework, this would imply that fewer constraints will be required to significantly improve the clustering accuracy. To this end, we developed a new method for actively selecting good pairwise constraints for semi-supervised clustering in the PCC framework [13].

Previous work in active learning has been mostly restricted to classification, where different principles ofquery selection have been studied, e.g., reduction of the version space size, reduction of uncertainty in predicted label, maximizing the margin on training data, and finding high variance data points by density-weighted pool-based sampling [14]. However, active learning techniques in classification are not applicable in the clustering framework, since the basic underlying concept of reduction of classification error and variance over the distribution of examples is not well-defined for clustering [15]. In the unsupervised setting, Hofmann et al. Consider a model of active learning which is different from ours – they have incomplete pairwise similarities between points, and their active learning goal is to select new data, using expected value of information estimated from the existing data, such that the risk of making wrong estimates about the true underlying clustering from the existing incomplete data is minimized [16]. In contrast, our model assumes that we have complete similarity information between all pairs of points, along with pairwise constraints whose violation cost is a component of the objective function, and the active learning goal is to select pairwise constraints which are most informative about the underlying clustering [17]. It also consider active learning in semi-supervised clustering, but instead of making example-level queries they make cluster level queries, i.e., they ask the user whether or not two whole clusters should be merged [1]. Answering example-level queries rather than cluster-level queries is a much easier task for a user, making our model more practical in a real-world active learning setting [2].

III.C1.Motivation Of Active Constraint Selection Algorithm

It was observed that initializing KMeans with centroids estimated from a set of labeled examples foreach cluster gives significant performance improvements [3]. Since good initial centroids are very critical for the successof greedy algorithms such as KMeans, we follow the same principle for the pairwise case: we will try to get as manypoints (proportional to the actual cluster size) as possible per cluster, so that PC-KMeans is initialized from a verygood set of centroids [4].In the exploration phase, we use a very interesting property of the farthest-first traversal [5]. Given a set of disjoint balls of unequal size in a metric space, we show that the farthest-first scheme is sure to get one point fromeach of the balls in a reasonably small number of attempts [6].

III.D.Unified Model Of Semi-Supervised Clustering

In previous work, similarity-based and search-based approaches to semi-supervised clustering have not been adequately compared experimentally, so their relative strengths and weaknesses are largely unknown [7]. Also, the two approaches are not incompatible; therefore, applying a search-based approach with a trained similarity metric is clearlyan additional option which may have advantages over both existing approaches [8]. In this work, we presented a newunified semi-supervised clustering algorithm derived from KMeans that incorporates both metric learning and usinglabeled data as seeds and/or constraints [9].

IV. Algorithms On Semi-Supervised Clustering

The key issue of high dimensional clustering is formulating a suitable mechanism tokeep functional harmony between dimensionality reduction and clustering [10]. Forclustering with unlabeled examples, although some unsupervised dimensionalityreduction or feature extraction methods such as PCA, independent componentanalysis, factor analysis can preserve the main information of a data set according totheir respective focuses, but their criteria are always not consist with the criterion of clustering, i.e., inherit cluster structures are destroyed severely in many cases [11]. Wethink that some information concerning class labeling may be required as a bridge toreach functional harmony for clustering problem [12]. Obviously, the cluster membershipcan be utilized instead of class labeling, but there is a dilemma whether to doclustering first or do dimensionality reduction first [13].In order to solve this problem, we propose a semi-supervised hierarchical clusteringalgorithm, in which the dimensionality is gradually reduced with a

semi-superviseddimensionality reduction algorithm as clusters are gradually formed [14]. The validity ofour method is based on the fact that the importance of the labeling information becomes more and more important as number of dimensions is gradually reduced [15].

IV.A. Semi-Supervised Dimensionality Reduction

Given a data set D = {(x1, y1), L, (x n, yn)} ∈R d × {class/cluster} and n is the total number of data, d is the number of original dimensions, x is the data vector, and y is the class label [16]. D can be divided into two sets D L U = { } ∪, where L is the set of labeled data with known class labels and U is the set of unlabeled data. The semi supervised dimensionality reduction algorithm is an optimization based on these two types of data, and the criterion of optimization is a combination of LDA and PCA that are popular dimensionality reduction techniques used for labeled and unlabeled data respectively [17]. Each example x in L belongs to a specific class Ci , and c is the number of classes; mi and pi are the mean vector and a priori probability of class Ci respectively; Sb and Sw are the between-class and within-class scatter matrices respectively [1]. The purpose of LDA is to maximize the between-class scatter while simultaneously minimizing the within-class scatter in the reduced l-dimensional space [2].

$$J_F(\boldsymbol{W}) = \frac{\boldsymbol{W}^T \boldsymbol{S}_b \boldsymbol{W}}{\boldsymbol{W}^T \boldsymbol{S}_w \boldsymbol{W}}$$

$$\boldsymbol{S}_b = \sum_{i=1}^{c} p_i (\boldsymbol{m}_i - \boldsymbol{m}_0)(\boldsymbol{m}_i - \boldsymbol{m}_0)^T$$

$$\boldsymbol{S}_w = \sum_{i=1}^{c} p_i E\{(\boldsymbol{x} - \boldsymbol{m}_i)(\boldsymbol{x} - \boldsymbol{m}_i)^T \,|\, \boldsymbol{x} \in C_i\}$$

$$\boldsymbol{m}_0 = \sum_{i=1}^{c} p_i \boldsymbol{m}_i$$

V. Clustering High Dimensional Data

The objects in data mining could have hundreds of attributes. Clustering in such high dimensional spaces presents tremendous difficulty, much more so than in predictive learning. In decision trees, for example, irrelevant attributes simply will not be picked for node splitting, and it is known that they do not affect Naïve Bayes as well [3]. In clustering, however, high dimensionality presents a dual problem. First, under whatever definition of similarity, the presence of irrelevant attributes eliminates any hope on clustering tendency [4]. After all, searching for clusters where there are no clusters is a hopeless enterprise. While this could also happen with low dimensional data, the likelihood of presence and number of irrelevant attributes grows with dimension [5].

The second problem is the dimensionality curse that is a loose way of speaking about a lack of data separation in high dimensional space [6]. Mathematically, nearest neighbor query becomes unstable: the distance to the nearest neighbor becomes indistinguishable from the distance to the majority of points. This effect starts to be severe for dimensions greater than 15 [7]. Therefore, construction of clusters founded on the concept of proximity is doubtful in such situations. For interesting insights into complications of high dimensional data [8]. Basic exploratory data analysis (attribute selection) preceding the clustering step is the best way to address the first problem of irrelevant attributes. We consider this topic in the section General Algorithmic Issues [9]. Below we present some techniques dealing with a situation when the number of already pre-selected attributes d is still high [10].

In the sub-section Dimensionality Reduction we talk briefly about traditional methods of dimensionality reduction [11]. In the sub-section Subspace Clustering we review algorithms that try to circumvent high dimensionality by building clusters in appropriate subspaces of original attribute space [12]. Such approach has a perfect sense in applications, since it is only better if we can describe data by fewer attributes [13]. Still another approach that divides attributes into similar groups and comes up with good new derived attributes representing each group is discussed in the sub-section Co-Clustering [14]. Important source of high dimensional categorical data comes from transactional (market basket) analysis. Idea to group items very similar to co-clustering has already been discussed in the section Co-Occurrence of Categorical Data [15].

V.A.Dimensionality Reduction

Many spatial clustering algorithms depend on indices in spatial datasets (sub-sectionData Preparation) to facilitate quick search of the nearest neighbors [16]. Therefore, indicescan serve as good proxies with respect to dimensionality curse performance impact.Indices used in clustering algorithms are known to work effectively

for dimensions below16 [17]. For a dimension d > 20 their performance degrades to the level of sequential search(though newer indices achieve significantly higher limits). Therefore, we can arguably claim that data with more than 16 attributes is high dimensional [1].
Two general purpose techniques are used to fight high dimensionality:
(1) Attributestransformations
(2) Domain decomposition.

Attribute transformations are simple functions of existent attributes. For sales profilesand OLAP-type data, roll-ups as sums or averages over time intervals (e.g., monthlyvolumes) can be used [2]. Due to a fine seasonality of sales such brute force approachesrarely work. In multivariate statistics principal components analysis (PCA) is popular, but this approach is problematic since it leads toclusters with poor interpretability [3]. Singular value decomposition (SVD) technique is usedto reduce dimensionality in information retrieval and statistics. Low-frequency Fourier harmonics in conjunctionwith Parseval's theorem are successfully used in analysis of time series, as well as wavelets and other transformations [4].

Domain decomposition divides the data into subsets, canopies, using some inexpensive similarity measure, so that the high dimensional computationhappens over smaller datasets [5]. Dimension stays the same, but the costs are reduced. Thisapproach targets the situation of high dimension, large data, and many clusters [6].

V.B.Subspace Clustering

Some algorithms better adjust to high dimensions. For example, the algorithm CACTUS (section Co-Occurrence of Categorical Data) adjusts well since it defines a cluster only in terms of a cluster's 2D projections [7]. In this section we cover techniques that are specifically designed to work with high dimensional data.

CLIQUE starts with the definition of a unit – elementary rectangular cell in a subspace. Only units whose densities exceed a threshold τ are retained. A bottom-up approach of finding such units is applied [8]. First, 1-dimensional units are found by dividing intervals in $\check{h}$ equal-width bins (a grid). Both parameters τ and $\check{h}$ are the algorithm's inputs. The recursive step from q-1-dimensional units to q-dimensional units involves self-join of q-1 units having first common q-2 dimensions (Apriori-reasoning) [9]. All the subspaces are sorted by their coverage and lesser-covered subspaces are pruned. A cut point is selected based on MDL principle. A cluster is defined as a maximal set of connected dense units [10].

It is represented by a DNF expression that is associated with a finite set of maximal segments (called regions) whose union is equal to a cluster. Effectively, CLIQUE results in attribute selection (it selects several subspaces) and produces a view of data from different perspectives [11]. The result is a series of cluster systems in different subspaces. This versatility goes more in vein with data description rather than with data partitioning: different clusters overlap [12]. If q is a highest subspace dimension selected, the complexity of dense unit's generations is. Identification of clusters is a quadratic task in terms of units [13].

The algorithm MAFIA (Merging of Adaptive Finite Intervals) significantly modifies CLIQUE. It starts with one data pass to construct adaptive grids in each dimension [14]. Many (1000) bins are used to compute histograms by reading blocks of data in core memory, which are then merged together to come up with a smaller number of variable-size bins than CLIQUE does [15]. The algorithm uses a parameter Į, called cluster dominance factor, to select bins that are Į-times more densely populated relative to their volume than on average. These are q=1 candidate dense units (CDUs) [16].

The algorithm OPTIGRID uses data partitioning based on divisive recursion by multi-dimensional grids. Authors present a very good introduction into the effects of high-dimension geometry [17]. Familiar concepts, as for example, uniform distribution, become blurred for large d. OPTIGRID uses density estimations in the same way the algorithm DENCLUE (by the same authors) does [1]. It primarily focuses on separation of clusters by (hyper) planes that are not necessarily axes parallel. To find such planes consider a set of contracting linear projectors (functional) P1...Pk, $Pj \leq 1$ (x - y) of the attribute space A at a 1D line [2].

The algorithm PROCLUS (Projected Clustering) associateswith a subset C a low-dimensional subspace such that the projection of C into thesubspace is a tight cluster. The subset – subspace pair when exists constitutes a projectedcluster [3]. The number k of clusters and the average subspace dimension l are user inputs.The iterative phase of the algorithm deals with finding k good medoids each associatedwith its subspace [4]. A sample of data is used in a greedy hill-climbing technique.Manhattan distance divided by the subspace dimension is a useful normalized metric forsearching among different dimensions [5]. An additional data pass follows after iterativestage is finished to refine clusters including subspaces associated with the medoids [6].

The algorithm ORCLUS (Oriented projected Cluster generation) uses a similar approach of projected clustering, but employs non-axes parallelsubspaces of high dimensional space [7]. In fact, both developments address a more genericissue: even in a low dimensional space, different portions of data could exhibit

clusteringtendency in different subspaces (consider several non-parallel non-intersecting cylindersin 3D space) [8].

Clustering Approach	Axis parallel	Grid based	Adaptive grid	Overlapping	Search Strategy		Use of Monotonicity property	Input Parameters	global density parameters	Shape of the Cluster	Robust to noise	Independent of order of data	Independent of order of dimensions	Arbitrary Subspace Dimensionality
					Bottom -up	Top-down								
CLIQUE	Y	Y	-	Y	Y	-	Y	ξ-grid interval τ-threshold	Y	F	N	Y	Y	Y
MAFIA	Y	Y	Y	Y	Y	-	Y	α-clus dominance factor	Y	F	Y	Y	Y	Y
DOC	Y	N	-	N	-	Y	N	ω-size of grid α-threshold β-bal bet'n pts & dims	Y	F	Y	Y	Y	Y
PROCLUS	Y	N	-	N	-	Y	-	k-no. of cluster l-avg no. of dims	N	F	Y	Y	Y	N
SUBCLU	Y	N	-	Y	Y	-	Y	ε-radius μ-threshold	Y	A	Y	Y	Y	Y
PreDeCon	Y	N	-	N		Y	Y	ε-radius μ-threshold λ, δ-preference para.s	Y	A	Y	Y	Y	N
FIRES	Y	N	-	Y	-	-	N	ξ-threshold σ-radius	N	F	Y	Y	Y	Y
DiSH	Y	N	-	Y	Y	-	Y	ξ-threshold σ-radius	Y	A	Y	Y	Y	Y
DENCLU	Y	Y	N	Y	Y	-	-	ξ-threshold σ-radius	-	A	N	Y	Y	N
OptiGrid	Y	Y	Y	Y	Y	-	-	ξ-threshold σ-radius	-	A	Y	Y	Y	Y

V.C. Co-Clustering

In OLAP attribute roll-ups can be viewed as representatives of the attribute groups. Aninteresting general idea of producing attribute groups in conjunction with clustering ofpoints themselves leads to the concept of co-clustering. Co-clustering is a simultaneousclustering of both points and their attributes [9]. This approach reverses the struggle: toimprove clustering of points based on their attributes, it tries to cluster attributes based onthe points [10]. So far we were concerned with grouping only rows of a matrix X. Now we aretalking about grouping its columns as well. This utilizes a canonical duality contained inthe point-by-attribute data representation [11].

The idea of co-clustering of data points and attributes is old and is known under the names simultaneous clustering, bi-dimensional clustering,block clustering, conjugate clustering, distributional clustering, and informationbottleneck method. The use of duality for analysis of categorical data (dual ormultidimensional scaling) also has a long history in statistics [12].

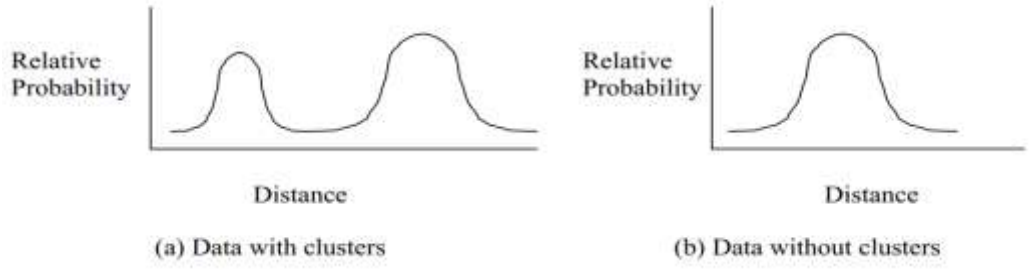
V.D. The "Curse Of Dimensionality"

It is the curse of dimensionality, a malediction that has plagued the scientist from the earliest days." The issue referred to in Bellman's quote is the impossibility of optimizing a function of many variables by a brute force search on a discrete multidimensional grid [13]. (The number of grids points increases exponentially with dimensionality, i.e., with the number of variables.) With the passage of time, the "curse of dimensionality" has come to refer to any problem in data analysis that results from a large number of variables (attributes) [14].

$$\lim_{d \to \infty} \frac{MaxDist - MinDist}{MinDist} = 0$$

In general terms, problems with high dimensionality result from the fact that a fixed number of data points become increasingly "sparse" as the dimensionality increase [15]. To visualize this, consider 100 points distributed with a uniform random distribution in the interval [0, 1]. If this interval is broken into 10 cells, then it is highly likely that all cells will contain some points [16]. However, consider what happens if we keep the number of points the same, but distribute the points over the unit square [17]. (This corresponds to the situation

where each point is two-dimensional.) If we keep the unit of discretization to be 0.1 for each dimension, then we have 100 two-dimensional cells, and it is quite likely that some cells will be empty [1].



VI. Recent Work In Clustering High Dimensional Data

Vi.A. Clustering Via Hypergraph Partitioning

Hyper graph-based clustering is an approach to clustering in high dimensional spaces, which is based on hyper graphs [2]. Hyper graphs are an extension of regular graphs, which relax the restriction that an edge can only join two vertices. The individual items are the vertices of the hyper graph. The hyper edges are determined by determining subsets of items that frequently occur together [3]. For example, baby formula and diapers are often purchased together [4]. These subsets of frequently co- occurring items are called frequent item sets and can be found using relatively simple and efficient algorithms [5]. The strength of the hyper edges is determined in the following manner. If the frequent item set being considered is of size n, and the items of the frequent item set are i1, i2… in, then the strength of a hyper edge [6].

VI.B. A "concept-based" approach to clustering highdimensional data

A key feature of some high dimensional data is that two objects may be highlysimilar even though commonly applied distance or similarity measures indicate that theyare dissimilar or perhaps only moderately similar [7]. Conversely, and perhapsmore surprisingly, it is also possible that an object's nearest or most similar neighborsmay not be as highly "related" to the object as other objects which are less similar [8]. Todeal with this issue we have extended previous approaches that define the distance or similarity of objects in terms of the number of nearest neighbors that they share [9]. Theresulting approach defines similarity not in terms of shared attributes, but rather in termsof a more general notion of shared concepts [10]. The rest of this section details our work infinding clusters in these "concept spaces," and in doing so, provides a contrast to theapproaches of the previous section, which were oriented to finding clusters in moretraditional vector spaces [11].

VII. Other Semi-Supervised Clustering Algorithms

We want to apply our semi-supervision ideas to hierarchical algorithms, e.g., HAC and Cobweb. Incorporating constraints into hierarchical algorithms will be relatively straightforward [12]. For example, to run constrained HAC, we can change the similarity then; we can proceed and run the usual HAC algorithm on the data points using this modified similarity metric, so that at each cluster-merge step, we consider the similarity between the data points as well as the cost of constraint violation incurred during the merge operation [13]. A more interesting problem would be when the initial supervision is given in the form of a hierarchy, and the clustering problem will be to do hierarchical clustering "using" the initial hierarchy [14]. We want to formalize the notion of using an initial seed hierarchy for hierarchical clustering [15]. Such an approach would be useful for content management applications, e.g., if the requirement is to hierarchically cluster the documents of a company, and the initial seed hierarchy is a preliminary directory structure containing a subset of the documents [16]. So far we have mainly focused on clustering algorithms that use a generative model. We also want to apply the pairwise constrained framework to discriminative clustering algorithms (e.g., graph partitioning), for which pairwise constraints are a natural way for providing constraints. Another interesting research direction would be online clustering in the semi-supervised framework [17].

VII.A.Ensemble Semi-Supervised Clustering

In our work so far, we have assumed constraints to be noise-free. We have also assumed the weights on the constraints to be uniform (PCKMeans) or changed the weights based on the "difficulty of satisfying the constraints" (unified model) [1]. An interesting problem in the PCC model would be the choice of the constraint weights in the general case of noisy constraints. Given a set of noisy constraints, we can create an ensemble of semi-supervised clusters, each of which put different weights on the constraints and possibly get different clustering's [2]. We propose a scheme for creating an ensemble of PCC clusters and combining their results using boosting.

Each PCC clustered can be considered as a weak learner taking pairwise data points as input, and giving an binary output decision of "same-cluster" or "different-cluster" [3]. The must-link and cannot-link constraints can be considered as the training data for each weak learner. Given a set of input constraints, the PCC clustered initially sets all constraints to have uniform weight and performs clustering [4]. After clustering is completed, the clustered categorizes each pair of points as "same-cluster" or "different-cluster", based on whether the pair ended up in the same cluster or in different clusters [5]. Since the given constraints are noisy, some of them will be violated by the clustering [6]. The constraints are reweighted based on the number of errors made by the weak learner, and a new clustered is created to perform the clustering with the new weights on the constraints. We use boosting for re-weighting of the constraints and combining the outputs of the clusters in the ensemble [7].

VIII. Clustering High Dimensions Data Techniques

The operations of clustering high dimensional data techniques have recently grown in advance [8].The popular methods asmentioned above were analyzed in detail.

VIII.A. Gaussian Mixture Models Using High-Dimensional Data

Clustering divides a given dataset {x1 , ..., xn } of n data points into k homogeneous groups. Popular clustering techniques use Gaussian Mixture Models (GMM), which assume that each class is represented by a Gaussian probability density. Data k{x1 , ..., xn } ∈ Rp are then modeled with the density f (x, θ) = ∑ where φ is a multi-variate normal density with parameter θi = {μi , Σi } and πi are the mixing proportions [9]. This model which uses to estimates full covariance matrices and therefore the number of parameters is very large in high dimensions [10].However, due to the empty space phenomenon we can assume that high-dimensional data live in subspaces with adimensionality lower than the dimensionality of the original space [11]. We here propose to the work in low-dimensional classspecific subspaces in order to adapt classification to high-dimensional data and to limit the number of parameters to estimate [12].

VIII.B.The Decision Rule

Classification assigns an observation x ∈ Rp with unknown class membership to one of k classes C1...Ck known a priori [13]. The optimal decision rule is the one which called Bayes decision rule, this affects the observation x to the class which has the maximum posterior probability P (x ∈ Ci |x) = πi φ(x, θi)/ l=1 πl φ(x, θl). Maximizing the posterior probability is equivalent to minimizing -2 log (πi φ(x, θi)) [14]. For the model [aij bi Qi di], this result in the decision rule δ + which assigns x to the class minimizing the following cost function Ki (x):

$$Ki(x)=||\mu i-Pi(x)||2Ai+ ||x-Pi(x)||2+\sum (aij)+(p-di)\log(bi)-2\log(\pi i)$$

Where ||.||Λi is the Mahalanobis distance associated with the matrix Λi =Qi ΔI Qit. The posterior probability can therefore be rewritten as follows: P (x ∈ Ci |x) = 1∑ (Ki (x) - Kl (x))). It measures the probability that x belongs to Ci2 and allowsto identify dubiously classified points [15]. We can observe that this new decision rule is mainly based on two distances: the distance between the projection of x on Ei and the mean of the class; and the distance between the observation and the subspace Ei [16]. This rule assigns a new observation to the class for which it is close to the subspace and for which its projection on the class subspace is close to the mean of the class [17]. If we consider the model [ai bi Qi di], the variances ai and bi balance the importance for the both distances. The example, if the data having too much noisy, i.e. bi is large, it is natural to balance the distance ||x – Pi (x) ||2 by 1/bi in order to take into account the large variance in E (1/i) [1].Remark that the decision rule δ+ of our models uses only the projection on Ei and we only have to estimate a di - dimensional subspace [2]. Thus, our models are significantly more parsimonious than the general GMM. For example, if we consider 100-dimensional data, that are made of 4 classes with common intrinsic dimensions di equal to 10, the model [ai bi Qi di] requires the estimation of 4 015 parameters whereas the full Gaussian mixturemodel estimates 20 303 parameters [3].

VIII.C. High Dimensional Data Clustering

In this section we derive the EM-based clustering framework for the model [aij bi Qi di] and the sub-models [4]. The new clustering approach are referred to by the High-Dimensional Data Clustering, which has the lack of space, we do not need to present the proofs of the following results which can be found in [5].

VIII.C1.The Clustering Method Hddc

Unsupervised classification organizes data in homogeneous groups using only the observed values of the p, whereas p is the explanatory variables [6]. Normally, the parameters use to estimate by the EM algorithm which repeats iteratively E or M steps. Suppose if we use the parameterization that presented in the previous

section, that the EM algorithm for estimating the parameters $\theta = \{\pi i , \mu i , \Sigma i , aij , bi , Qi , di \}$, would be written as follows [7]:

E step: this step computes at the iteration q the conditional posterior probabilities: tij(b)=P(xj£Ci(q)|xj),from the relation , it may consider:

tij(q)=1/∑ (1/2(Ki(q-1)(xj)-Kl(q-1)(xi))) (1)
Where Ki is defined

M step: this step maximizes at the iteration q has the conditional likelihood [8]. Proportions,which means and covariance matrices of the mixture are estimated by:
πi(q)=(ni(q) /n), μi(q)=(1/ni(q))∑ ij(q)xj,ni(q)=∑ ij(q) (2)
∑ (1/ni(q))∑ ij(q)(xj-μi(q))(xj-μj(q))t (3)
The estimation of the HDDC parameters is detailed in the following subsection.

VIII. Conclusion

We have provided a brief introduction to cluster analysis with an emphasis on the challenge of clustering high dimensional data [9]. The principal challenge in extending cluster analysis to high dimensional data is to overcome the "curse of dimensionality," and we described, in some detail, the way in which high dimensional data is different from low dimensional data, and how these differences might affect theprocess of cluster analysis [10]. Finally, high dimensional data is only one issue that needs to be considered when performing cluster analysis. In closing we mention some other, only partially resolved, issues in cluster analysis: scalability to large data sets, independence of the order of input, effective means of evaluating the validity of clusters that are produced, easy interpretability of results, an ability to estimate any parameters required by the clustering technique, an ability to function in an incremental manner, and robustness in the presence of different underlying data and cluster characteristics [11]. Subspace clustering algorithms help solve the problems of clustering in high dimensional data by using different techniquesto locate clusters in different subsets of the complete dimension set [12]. Density based clustering algorithms perform better, compared to other subspace clustering approaches, by generating clusters of adaptive size, shape, densities and dimensionalities [13]. In this survey various techniques of Cluster high dimensional data were described in detail. These techniques are most important which uses to find the similar functionality at genes and proteins [14]. The Clustering high dimensional data techniques mentioned in this review paper are used in many advanced for summarization or improved understandings. This high dimensionaldata in clustering is to determine the intrinsic grouping in a set of unlabeled data [15].

Lot of such approaches exists for subspace clustering and numerous algorithms are being proposed nearly every day. Proper selection of a clustering approach to suit a particular application and data should be based on the understanding of the exact requirement of clustering application and the principles ofworking of available approaches [16]. Hence, in this paper an attempt is made to present various density based subspace clustering algorithms to better understand their comparative characteristics [17]. A comparative chart is prepared on the basis of various performance parameters and presented for a ready reference. We hope, this will surely help future developers to select a set of relevant /appropriate approaches from the given list, against which developers can test / compare the results of their proposed subspace clustering algorithm [1]. Finally, we limited the scope of this paper only to few, significant representative contributions and that too clustering based on continuous valued data [3]. There exist many clustering algorithms which are specially designed for stream data, graph data, spatial data, text data, heterogeneous data etc [2]. We hope to stimulate further research in these areas. From above these contents we can conclude that there are various methods we can use to form cluster in semi supervised clustering [4]. Each method has it's own some benefits and limitations. For constant dataset all methods are ok ,but for updated data incremental semi supervised clustering would be more useful, because in this the data is continuously entered in system, continuously update data, and form new clusters as per their contents, and sometimes changes clusters as per user demands [5]. This data is labeled or unlabeled or in shape so incremental can work on all these type of data than other methods. So incremental semi supervised clustering is can be used method of clustering approach [6].

The proposed algorithm is based on semi-supervised hierarchicalclustering frame in which the clusters are formed gradually from a small amount oflabeled examples as seeds by assigning unlabeled examples to the existed clustersaccording to their distances [10]. In the hierarchical clustering procedure, dimensionalityreduction is incorporated, and the number of dimensions is reduced gradually as thefinal clusters are formed [7]. The criterion of dimensionality reduction is dependent onboth the labeled data in the current clusters and the unlabeled data that have not beenassigned to the current clusters [9]. The purpose of this article is to present a comprehensive classification of different clustering techniques for high dimensional data. Clustering high dimensional data sets is a ubiquitous task [8]. The incosent growth in the fields of

communication and technology, there is tremendous growth in high dimensional data spaces. It study focuses on issues and major drawbacks of existing algorithms [11]. As the number of dimensions increase, many clustering techniques begin to suffer from the curse of dimensionality, de-grading the qualityof the results [12]. In high dimensions, data becomes very sparse and distance measures become increasinglymeaningless. This problem has been studied extensively and there are various solutions, each appropriate fordifferent types of high dimensional data and data mining procedures [13].

Future Enhancement

The Iterative Framework requires repeated re-clustering of the data with an incrementally growing constraint set [14]. This can be computationally demanding for large data sets. To address this problem, it would be interesting to consider an incremental semi supervised clustering method that updates the existing clustering solution based on the neighborhood assignment for the new point [15]. An alternative way to lower the computational cost is to reduce the number of iterations by applying a batch approach that selects a set of points to query in each iteration [16]. A naive batch active learning approach would be to select the top k points that have the highest normalized uncertainty to query their neighborhoods [17]. However, such a strategy will typically select highly redundant points. Designing a successful batch method requires carefully trading off the value (normalized uncertainty) of the selected points and the diversity among them [1]. In our future work we will investigate the sensitivity of our approach with respect to the dimensionality of subspaces, and possibly define an heuristic to automatically estimate an "optimal" value for such parameter [2]. Furthermore, we will explore alternative mechanisms to credit weights to features by utilizing the constraints; consequently we will bias the sampling in feature space to favor the estimated most relevant features [3].

Through the iterative clustering – dimensionalityreduction - clustering procedure, the harmony between clustering and dimensionalityreduction is reached, and these two tasks are integrated into a harmonious system [4]. Theexperimental results also demonstrate the effectiveness of our method [5]. However, howto automatically determine suitable values for the parameters in our methods, and howto improve the computational effectiveness for large scale data sets, are need to befurther studied in the future [6]. As with any clusteringtechniques, finding meaningful and useful resultsdepends on the selection of the appropriate clusteringtechnique [7]. In order to do this, one must understand thedataset in a domain specific context in order to be ableto best evaluate the results from various approaches [8].From the above discussion it is observed that the currenttechniques will suffers with many problems [9]. To improvethe performance of the data clustering in highdimensional data, it is necessary to perform research in the areas like dimensionality reduction, redundancyreduction in clusters and data labeling [10]. Finding connections and sharing ideas among these related topics will likely not only yield interesting future research directions, but alsohelp resolve many challenges in high-dimensional data visualization [11].

References

[1]. S. Basu, A. Banerjee, and R. Mooney, "Active Semi-Supervision for Pairwise Constrained Clustering," Proc. SIAM Int'l Conf. Data Mining,pp.333-344, 2014.

[2]. H. Zeng and Y.-M.Cheung, "Semi-supervised maximum margin clustering with pairwise constraints," IEEE Trans. Knowl. Data Eng., vol. 24,no. 5, pp. 926–939, May 2012.

[3]. R. Agrawal, J. Gehrke, D. Gunopulos and Raghavan," Automatic subspace clustering of high dimensional data fordata mining applications", In Proceedings of the SIGMOD,Vol. 27 Issue 2, pp. 94-105,2012.

[4]. J. M. Pena, J. A. Lozano, P. Larranaga and Inza, I," Dimensionality reduction in unsupervised learning of conditional gaussian networks", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23(6):590 – 603, 2011.

[5]. Goil, S., Nagesh, H. and Choudhary, A., "MAFIA: Efficient and scalable subspace clustering for very large datasets", Northwestern University, Technical Report CPDC-TR-9906-010, 2010.

[6]. A. Hinneburg and D. A. Keim, "Optimal grid clustering: Towards breaking the curse of dimensionality in highdimensional clustering," In Proceedings of 25th International Conference on Very Large Data Bases (VLDB-2013), pp.506-517, Edinburgh, Scotland, September, Morgan Kaufmann, 2013.

[7]. I. Assent, R. Krieger, E. Muller, and T. Seidl, "DUSC: Dimensionality Unbiased Subspace Clustering". In Proc.IEEE Intl. Conf. on Data Mining (ICDM 2014), Omaha, Nebraska, pp 409-414, 2014.

[8]. George Karypis and Vipin Kumar ," A hypergraph partitioning package", Technical report, Department of Computer Science, University of Minnesota ,2010.

[9]. HarashaNagesh, Sanjay Goil, and AlokChoudhary, "MAFIA: Efficient and Scalable Subspace Clustering for Very Large Data Sets," Technical Report Number CPDC-TR-9906-019, Center for Parallel and Distributed Computing, Northwestern University, 2011.

[10]. Jacob Kogan, "Introduction to Clustering Largeand High-Dimensional Data", University of Maryland, Baltimore County, 2010.

[11]. Chris Ding Xiaofeng He, "Adaptive dimensionreduction for clustering high dimensional data ",in the Proceedings of IEEE InternationalConference on Data Mining, Washington DC, USA, 2012.

[12]. Lance Parsons, EhteshamHaque and Huan Liu ,"Subspace Clustering for High Dimensional Data: A Review", in the proceedings of SIGKDDExplorations, Volume 6, Issue 1, pages 90-105,2011.

[13]. M. Ankerst, M. M. Breunig, H.P. Kriegel, and J. Sander."OPTICS: Ordering Points to Identify the Clustering Structure". InProc. ACM SIGMOD Int. Conf. on Management of Data (SIGMOD'09), 2009.

[14]. Alijamaat, M. Khalilian, and N. Mustapha, "A NovelApproach for High Dimensional Data Clustering," Third International Conference on Knowledge Discoveryand Data Mining, pp. 264-267, Jan. 2010.
[15]. M. Hassani, Y. Kim, S. Choi, and T. Seidl, "Subspace clustering of data streams: new algorithms and effective evaluation measures," Journal ofIntelligent Information Systems, Jun. 2014.
[16]. W. Liu and J. OuYang, "Clustering algorithm for high dimensional datastream over sliding windows." IEEE, pp. 1537–1542,Nov. 2011.
[17]. H.-P. Kriegel, P. Krger, and A. Zimek, "Clustering high-dimensionaldata: A survey on subspace clustering, pattern-based clustering, andcorrelation clustering," ACM Trans. Knowl. Discov. Data, vol. 3, no. 1,pp. 1:1–1:58, Mar. 2009.

M. Pavithra" A Survey on Semi Supervised Clustering For High Dimensional Data Clustering" International Journal Of Engineering Science Invention (Ijesi), Vol. 08, No. 03, 2019, Pp 52-64