SCALABILITY ANALYSIS THROUGH AGGLOMERATIVE (HIERARCHICAL) CLUSTERING – COMPLETE AND CENTROID ALGORITHM

Ms.A.Uma Maheswari M.C.A.,M.Phil[1], Dr.N.Revathy MCA.,M.Phil.,Ph.D[2]
[1]Ph.D Research Scholar, [2]Associate Professor
Department of Master of Computer Applications,
Hindusthan College of Arts and Science, Coimbatore, India

Abstract: Clustering is a well-known problem in statistics and engineering, namely, how to arrange a set of vectors (measurements) into a number of groups (clusters). Clustering is an important area of application for a variety of fields including data mining, statistical data analysis and vector quantization. The problem has been formulated in various ways in the machine learning, pattern recognition optimization and statistics literature. The fundamental clustering problem is that of grouping together (clustering) data items that are similar to each other. The most general approach to clustering is to view it as a density estimation problem. Classification algorithms rely on human supervision to train it to classify data into pre-defined categorical classes. The term "classification" is frequently used as an algorithm for *all* data mining tasks. Instead, it is best to use the term to refer to the category of supervised learning algorithms used to search interesting data patterns. While classification algorithms have become very popular and ubiquitous in DM research, it is just but one of the many types of algorithms available to solve a specific type of DM task.

Keywords - Clustering, Scalability, Agglomerative clustering.

INTRODUCTION

The DM and KDD fields are relatively new; different authors appear to survey methods in different ways.

- Fayyad's methods for data mining: Predictive Modeling; Clustering; Summarization; Dependency Modeling; Change and Deviation Detection
- Goebel & Grunewald's methods for data mining: Statistical Models; Case-Based Reasoning; Neural Networks; Decision Trees; Rule Induction; Bayesian Belief Networks; Genetic algorithms; Fuzzy Sets; Rough Sets
- Aggarwal & Yu's survey of techniques for data mining: Association rules, Clustering, Classification

In the course of gathering information for this thesis, Aggarwal and Yu's survey turned out to be very helpful in organizing descriptions of various methods related with data mining. So in this dissertation we classify and approach these techniques as per the Aggarwal and Yu's survey on this field.

PROBLEM DEFINITION

Clustering is a well-known problem in statistics and engineering namely, how to arrange a set of vectors (measurements) into a number of groups (clusters). Clustering is an important area of application for a variety of fields including data mining, statistical data analysis and vector quantization. The problem has been formulated in various ways in the machine learning, pattern recognition optimization and statistics literature. The fundamental clustering problem is that of grouping together (clustering) data items that are similar to each other. The most general approach to clustering is to view it as a density estimation problem. Because of its wide application, several algorithms have been devised to solve the problem. Notable among these are the EM algorithm, neural nets, SVM and k-means. Clustering the data acts as a way to parameterize the data so that one does not have to deal with the entire data in later analysis, but only with these parameters that describe the data. Sometimes clustering is also used to reduce the dimensionality of the data so as to make the analysis of the data simpler.

In one of its forms, clustering problems can be defined as: given a dataset of N records, each having dimensionality d, to partition the data into subsets such that a specific criterion is optimized. The most widely used criterion for optimization is the distortion criterion. Each record is assigned to a single cluster and distortion is the average squared Euclidean distance between a record and the corresponding cluster center. Thus this criterion minimizes the sum of the squared distances of each record from its corresponding center.

Classification algorithms rely on human supervision to train itself to classify data into pre-defined categorical classes. For example, given classes of patients that corresponds to medical treatment responses; identify most responsive forms of treatment for the patient.

The following list shows some of the categories of classification algorithms generally used in data mining applications. In this dissertation those categories of algorithms are going to be addressed in detail. In this dissertation a detailed survey on existing algorithms will be made and the scalability of some of the existing classification algorithms will be examined.

- *k*-Nearest Neighbor algorithm
- Decision Tree.
- DNF Rules
- Neural networks
- Genetic algorithms
- Bayesian networks
- Rough and Fuzzy Sets

The term "classification" is frequently used as an algorithm for *all* data mining tasks [4]. Instead, it is best to use the term to refer to the category of supervised learning algorithms used to search interesting data patterns. While classification algorithms have become very popular and ubiquitous in DM research, it is just but one of the many types of algorithms available to solve a specific type of DM task [1].

Scalability refers to the ability of data mining algorithms to work under increasingly large databases. Because data mining deals with large databases, scalability is a desirable feature.

Literally, scalability means that as a system gets larger, its performance improves correspondingly [5]. For data mining, scalability means that by taking advantage of parallel database management systems and additional CPUs, you can solve a wide range of problems without the need to change your underlying data mining environment. You can work with more data, build more models, and improve their accuracy by simply adding additional CPUs. Ideally, scalability should be linear or better. For example, if you double the number of CPUs in a parallel system, you can build twice as many models in the same amount of time, or the same number of models in half the time.

A REVIEW ON DATA MINING AND KNOWLEDGE DISCOVERY

Data mining has been the subject of many recent articles in business and software magazines. However, just a few short years ago, few people had not even heard of the term data mining. Though data mining is the evolution of a field with a long history, the term itself was only introduced relatively recently in the 1990s.

The roots of data mining can be traced back along three family lines. The longest of these three is classical statistics. Without statistics, there would be no data mining, as these statistics are the foundation of most technologies on which data mining is founded upon. Classical statistics embrace concepts such as regression analysis, standard distribution, standard deviation, standard variance, cluster analysis, and confidence intervals, all of which are used primarily to study data and data relationships. These are the very building blocks on which more advanced statistical analyses are built upon. Even in today's data mining tools and knowledge discovery techniques, classical statistical analysis still plays a significant role.

The second longest family line for data mining is artificial intelligence, AI. This discipline, which is built upon heuristics as opposed to statistics, attempts to apply human thought-like processing to statistical problems. Because this approach requires vast amounts of computer processing power, it was not practical until the early 1980s, when computers began to offer useful power at reasonable prices [13]. AI found

relatively few applications at the very high-end scientific and government markets, and the required supercomputers of the era priced AI out of the reach of virtually everyone else [13]. The notable exceptions were certain AI concepts that were adopted by some high-end commercial products, such as query optimization modules for Relational Database Management Systems. Over the time, this changed, as AI was used to create new ways in addressing and solving very complex and math-driven problems. At the Artificial Intelligence Laboratory at MIT, founded in the 1960s, there is extensive research in many aspects of intelligence. Their aim is two-fold: to understand human intelligence at all levels, including reasoning, perception, language, development, learning, and social levels; and to build useful artifacts based on intelligence.

The third family line of data mining is machine learning, which is more accurately described as the union of statistics and AI. While AI was not a commercial success, and is therefore primarily used as a research tool, its techniques were largely co-opted by machine learning. Machine learning, able to take advantage of the ever-improving price/performance ratios offered by computers of the 1980s and 1990s, found more applications because the entry price was lower than AI. Machine learning could be considered an evolution from AI because it blends AI heuristics with advanced statistical analysis. Machine learning attempts to let computer programs learn about the data they study, such that programs make different decisions based on the characteristics of the studied data, using statistics for fundamental concepts, and adding more advanced AI heuristics and algorithms to achieve its goals.

As such, data mining, in many ways is fundamentally the adaptation of machine learning techniques to business applications. Data mining is best described as the union of historical and recent developments in statistics, AI, and machine learning. These techniques are used together to study data and find previously hidden trends or patterns within. Data mining is finding increasing acceptance in science and business areas that need to analyze large amounts of data to discover trends that they could not otherwise find.

KNOWLEDGE DISCOVERY PROCESS

Data mining is part of a larger iterative process called knowledge discovery. The following summarizes the steps of the knowledge discovery process.

- Define the Problem. This initial step involves understanding the problem and figuring out what the goals and expectations are of the project.
- Collect, clean, and prepare the data. This requires figuring out what data are needed, which data are most important and integrating the information. This step requires considerable effort, as much as 70% of the total data mining effort [14].
- Data mining. This model-building step involves selecting data mining tools, transforming the data if the tool requires it, generating samples for training and testing the model, and finally using the tools to build and select a model.
- Validate the models. Test the model to ensure that it is producing accurate and adequate results.
- Monitor the model. Monitoring a model is necessary as with passing time, it will be necessary to revalidate the model to ensure that it is still meeting requirements. A model that works today may not work tomorrow and it is therefore necessary to monitor the behavior of the model to ensure it is meeting performance standards.

THE DATA MINING PROCESS

The goal of identifying and utilizing information hidden in data has three requirements [1]:

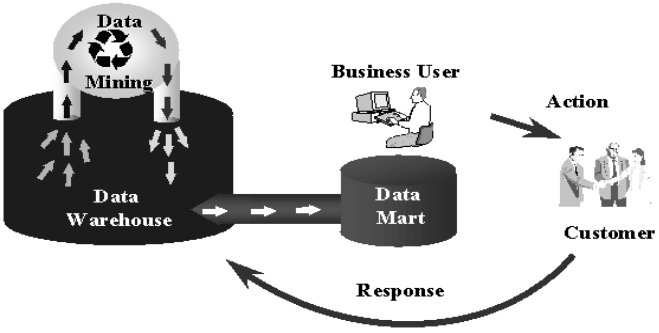
- The captured data must be integrated into organization-wide views instead of specific views.
- The information contained in the integrated data must be extracted.
- The obtained information must be organized in ways that enable decision-making.

The data mining process can be classified as going through a series of four steps. This consists of transforming the already summarized data found in a data warehouse into information than can produce useful results. These four steps can be summarized into [1]:

- Data selection
- Data transformation
- Mining the Data
- Interpretation of Results

Data selection consists of gathering the data for analysis. Data transformation will then convert appropriate data to a particular format. Data mining will then extract the desired type of information yielding in results to be interpreted. In Figure 1, the data-mining tool will extract the relevant information from the data warehouse environment. In order for the data-mining tool to work, the sub-processes of data selection and

transformation must take place prior to data mining. The results are then passed to a decision-oriented databases or data mart, where the user can make a recommendation based on the results and put the recommendations into action. Of course this assumes that all of the four steps will be successfully completed, which is not always the case.



Data Mining Process

Data selection can be the most important step in the process. This is due to the complexity in finding and constructing pre-selection criteria before the extraction of data actually transpires. The variables selected and the range of each of them should be determined in this step. For example, a marketing executive wishing to improve sales figures will pre-select those customers that have been most active in making purchases and observe their behavior. An executive can mine all the data, but this can turn out to be a very costly operation because the data-mining tool will have to search through all this data and moreover if results are generated, they have more risk in predicting an optimal recommendation. Carefully choosing the data is therefore a very important step.

Once the data to be mined has been chosen the next step in the data mining process usually consists of transforming the data into the particular formats needed by the data-mining tool. Data are further synthesized by computing certain rations and applying algorithms to convert the data to a particular type suitable for future applied tools [1].

Once the data have been selected and required transformations done, a data-mining tool can now be applied. Specific predictions about futuristic events based on previous collected data can yield in significant hidden findings through the use of well designed algorithms, a topic of discussion in later sections. Using a data warehouse alongside with a mining tool is usually recommended as this allows for a more efficient organization of the collected data in ways that can facilitate and optimize analysis. Furthermore, the mining tool can also interface with a DSS for further interpretation of the data.

However, a data mining system need not always interact with a data warehouse, and in fact, data mining can still extract pertinent information if given raw data from a database. The main advantage of using a data warehouse is that most of the data are already integrated in a suitable format of choice making it easier for a data-mining tool to extract the higher quality information.

The final step in the data mining process consists of interpreting the results. Once the extracted information is analyzed and interpreted, the most relevant information can be passed onto the decision-maker through a DSS. Result interpretation can consist not only of interpreting the output but also of further filtering the data and passing that information to the decision support system. In the case that the interpreted results are not satisfactory, it may be necessary to repeat any of the previous steps until the information generated contains the maximum added value to the data miner.

As such, data mining is a very complex process. Many steps need to be performed correctly before feeding of data to the data mining tool. Furthermore it is not guaranteed that the data-mining tool will yield significant results in any steps of the mining process. Certainly, performing many trials are recommended as this can reveal error corrections in any of the four steps. Any of the previously mentioned steps can be modified to continue investigating the data and searching for hidden patterns. This is the challenge of the data mining organization and though it can be a painstaking process, the more data that is mined, the more likely the data miner will learn from the process.

The use of tools such as DSS and a warehouse environment complement the data mining tools used to find useful facts buried in layers of data. To maximize the efficiency of data mining, both of these other tools must provide high quality delivery information to the data-mining tool. The use of good complementary tools to sift through data along with a powerful data-mining tool should be part of a well designed environment [1].

KNOWLEDGE DISCOVERY IN DATABASES

Due to the explosive growth of information available in databases, researchers have been looking for ways to make use of data stored. Knowledge Discovery in Databases is the process of finding useful information in a large database. Many algorithms have been suggested and used to find useful data, and the activity is called data mining. The purpose of Knowledge Discovery in Database is to find useful information in an ever growing data set size in databases. Computer assisted transactions have generated mountains of data that could be useful to maintain the competitive edge of a company. For instance, retailers such as Wal-Mart have been making the best use of information from their 20 million per day point-of-sale transactions, to understand customer purchasing behavior and improve customer service.

While the terms "Data mining" (DM) and "Knowledge Discovery in Databases" (KDD) have been used interchangeably, it is best to differentiate these terms since DM is one of the stages of KDD. The KDD process is generally made up of the following iterative stages:

- Understand the business: This phase is used to identify the domain, objectives, requirements, and feasibility of the data mining project.
- Understand data: This phase is based on data mining objectives identified in the previous stage. It is used to analyze and document available data and knowledge sources in the business, and to study characteristics of the data.
- Prepare data: Format or transform data to suitable media, clean or eliminate missing tuples, and focus the data for data mining. This step is usually the most time consuming part of the KDD project.
- Explore data: This phase is used to explore interesting insights into the data, so that initial hypotheses and models of data can be evaluated and developed.
- Data mining: Select and apply modeling and discovery algorithms to find patterns of interest in the data. This is accomplished by using different algorithms, which is the subject of this paper.
- Evaluate results: Interpret and evaluate data mining results in business terms, and initiate new experiments if necessary.
- Deploy results: Implement results obtained by the data mining process. Maintenance and monitoring data mining results are also part of this phase.
- Document experience: This phase is really done throughout all phases of KDD projects, and documents aspects of the project and deployment of data mining results. This stage enables future projects to be more efficient and effective.

The term "classification" is frequently used as an algorithm for all data mining tasks. Instead, it is best to use the term to refer to the category of supervised learning algorithms used to search interesting data patterns. While classification algorithms have become very popular and ubiquitous in DM research, it is just but one of the many types of algorithms available to solve a specific type of DM task.

Scalability refers to the ability of data mining algorithms to work under increasingly large databases. Because data mining deals with large databases, scalability is a desirable feature.

DATA MINING TASKS

Based on the data collected, data mining algorithms are used to either produce a description of the data stored, or predict an outcome. Different kinds of algorithms are used to achieve either one of these tasks. However, in the overall KDD process, any mixture of these tasks may be called upon to achieve the desired results [17]. For example, in determining consumer preference for a new product, an analyst might first use clustering to segment the customer database, and then apply regression to predict buying behavior for that cluster.

- Description tasks: These tasks describe the data being mined and they are:
  - Summarization: To extract compact patterns that describes subsets of data. The method used to achieve this task is Association Rule algorithms.
  - Segmentation or Clustering: To separate data items into subsets those are similar to each other. Partition-based clustering algorithms are used to achieve this task.
  - Change and Deviation Detection: To detect changes in sequential data (such as protein sequencing, behavioral sequences, etc.).
  - Dependency Modeling: To construct models of causality within the data.
- Prediction tasks: To predict some field(s) in a database based on information in other fields.
  - Classification: To predict the most likely state of a categorical variable (its class).
  - Regression: To predict results that is numeric continuous variables.

A REVIEW ON CLUSTERING AND CLASSIFICATION ALGORITHMS

The classification of large data sets is an important problem in data mining. The classification problem can be simply stated as follows. For a database with a number of records and for a set of classes such that each record belongs to one of the given classes, the problem of classification is to decide the class to which a given record belongs. The classification problem is also concerned with generating a description or model for each class from the given data set. Classification is a supervised learning [4]. Here a training data set of records is accompanied by class labels. New data can be classified based on the training set by generating descriptions of the classes. In addition to the training set, there is also a test data set which is used to determine the effectiveness of a classification. There are several approaches to classification. Decision trees are especially attractive in the data mining environment as they represent rules. Rules can be easily expressed in natural language and are easily comprehensible.

THE ALGORITHM UNDER EVALUATION

STAGES IN A CLUSTERING TASK

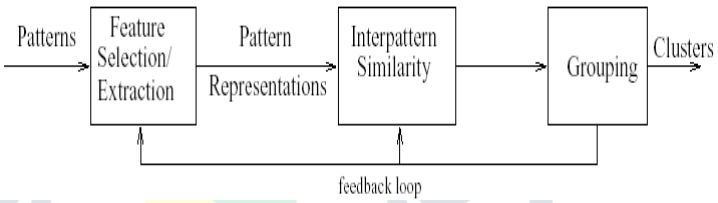
Figure below represents the typical sequencing of clustering activity [1]. A *pattern* (or *feature vector*) is a single data item that is used by clustering algorithm. It typically consists of a vector of *d* measurements

$$\mathbf{x} = (x_1, \ldots x_d)$$

(where *d* the dimensionality of the data):.

The individual scalar components x_i *of* a patter X is are called features (or attributes). Pattern representation refers to the number of classes, the number of available patterns, the number, type and scale of the features available to clustering algorithms.

Feature selection is the process of identifying the most effective subset of 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 what is called a *feature set* (or feature vector).



Stages in Clustering

Pattern proximity is usually measured by a distance function defined on pairs of patterns. A variety of distance functions are in use in various communities. A simple distance measure can often be used to reflect dissimilarity between two patterns, where other similarity measures can be used to characterize the conceptual similarity between two patterns. The Euclidian distance metric can be defined as follows:

$$d_2(\mathbf{x}_i, \mathbf{x}_j) = (\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2)^{1/2}$$

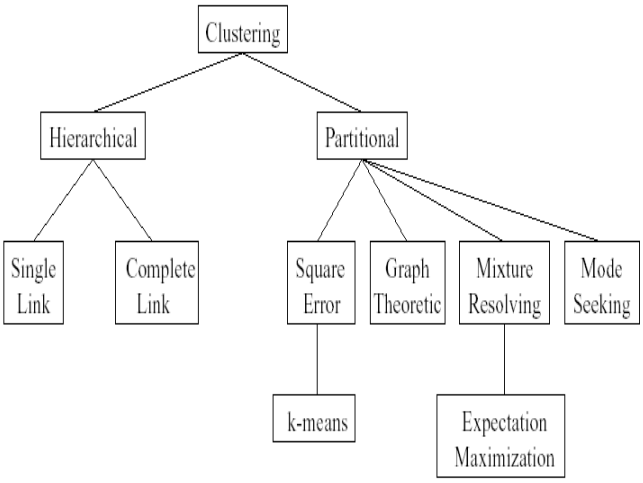
$$= \|\mathbf{x}_i - \mathbf{x}_j\|_2,$$

Where X_1 and X_2 are two patterns. Euclidian distance metric works well when the data set has "compact" or "isolated" clusters.

Another class of metrics characterizes conceptual similarity between two patterns. For example in Conceptual cluster (which we don't discuss in this paper), the similarity between X_1, X_2 is defined as

Where $\mathscr{C}$ is a $s(\mathbf{x}_i, \mathbf{x}_j) = f(\mathbf{x}_i, \mathbf{x}_j, \mathscr{C}, \mathscr{E})$, set of pre-defined concepts.

The Grouping step represents the organization of patterns into clusters based on pattern similarity. There are many clustering methods available, and each of them may give a different grouping of a dataset. The choice of a particular method will depend on the type of output desired, the known performance of method with particular types of data, the hardware and software facilities available and the size of the dataset. The taxonomy of clustering algorithms can be seen in the *figure.*

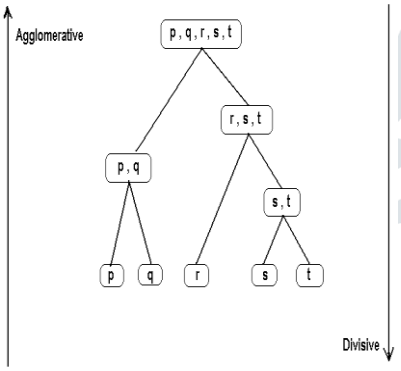


Taxonomy of Clustering Approaches

In general, clustering methods may be divided into two categories based on the cluster structure, which they produce. The non-hierarchical methods divide a dataset of N objects into M clusters, with or without overlap.

These methods are sometimes divided into *partitioning* methods, in which the classes are mutually exclusive, and the less common *clumping* method, in which overlap is allowed. Each object is a member of the cluster with which it is most similar; however the threshold of similarity has to be defined. The hierarchical methods produce a set of nested clusters in which each pair of objects or clusters is progressively nested in a larger cluster until only one cluster remains. The hierarchical methods can be further divided into *agglomerative* or *divisive* methods. In *agglomerative* methods, the hierarchy is build up in a series of N-1 agglomerations, or Fusion, of pairs of objects, beginning with the un-clustered dataset. The less common *divisive* methods begin with all objects in a single cluster and at each of N-1 steps divide some clusters into two smaller clusters, until each object resides in its own cluster.

HIERARCHICAL CLUSTERING



Hierarchical Clustering

In hierarchical clustering the data are not partitioned into a particular cluster in a single step. Instead, a series of partitions takes place, which may run from a single cluster containing all objects to n clusters each containing a single object. Hierarchical Clustering is subdivided into *agglomerative* methods, which proceed by series of fusions of the n objects into groups, and *divisive* methods, which separate n objects successively into finer groupings. Hierarchical clustering may be represented by a two dimensional diagram known as dendrogram which illustrates the fusions or divisions made at each successive stage of analysis. An example of such a dendrogram is given below:

AGGLOMERATIVE METHODS

An agglomerative hierarchical clustering procedure produces a series of partitions of the data, P_n, P_{n-1}, ..., P_1. The first P_n consists of n single object 'clusters', the last P_1, consists of single group containing all n cases.

At each particular stage the method joins together the two clusters which are closest together (most similar). (At the first stage, of course, this amounts to joining together the two objects that are closest together, since at the initial stage each cluster has one object.)
Differences between methods arise because of the different ways of defining distance (or similarity) between clusters. Several agglomerative techniques will now be described in detail.

SINGLE LINKAGE CLUSTERING

One of the simplest agglomerative hierarchical clustering method is single linkage, also known as the nearest neighbor technique. The defining feature of the method is that distance between groups is defined as the distance between the closest pair of objects, where only pairs consisting of one object from each group are considered.

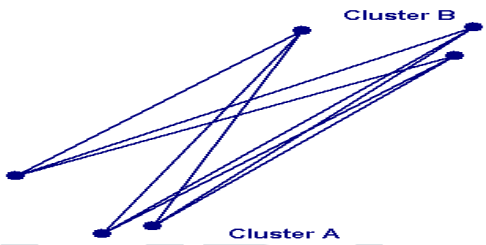
In the single linkage method, ***D(r,s)*** is computed as

D(r,s) = Min { *d(i,j)* : Where object *i* is in cluster ***r*** and object *j* is cluster ***s*** }

Here the distance between every possible object pair *(i,j)* is computed, where object *i* is in cluster ***r*** and object *j* is in cluster ***s***. The minimum value of these distances is said to be the distance between clusters ***r*** and ***s.*** In other words, the distance between two clusters is given by the value of the shortest link between the clusters.

At each stage of hierarchical clustering, the clusters ***r*** and ***s***, for which ***D(r,s)*** is minimum, are merged.

This measure of inter-group distance is illustrated in the figure below:



Complete Linkage Clustering

The complete linkage, also called farthest neighbor, clustering method is the opposite of single linkage. Distance between groups is now defined as the distance between the most distant pair of objects, one from each group. In the complete linkage method, ***D(r,s)*** is computed as

D(r,s) = Max { *d(i,j)* : Where object *i* is in cluster ***r*** and object *j* is cluster ***s*** }

Here the distance between every possible object pair *(i,j)* is computed, where object *i* is in cluster ***r*** and object *j* is in cluster ***s*** and the maximum value of these distances is said to be the distance between clusters ***r*** and ***s.*** In other words, the distance between two clusters is given by the value of the longest link between the clusters. At each stage of hierarchical clustering, the clusters ***r*** and ***s***, for which ***D(r,s)*** is minimum, are merged. The measure is illustrated in the figure below:

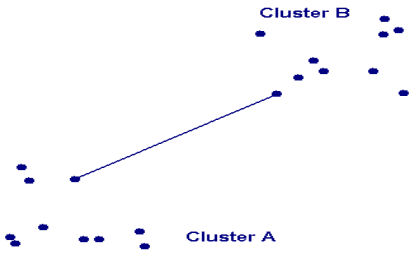
Average Linkage Clustering

Here the distance between two clusters is defined as the average of distances between all pairs of objects, where each pair is made up of one object from each group.

In the average linkage method, ***D(r,s)*** is computed as

$$D(r,s) = T_{rs} / (N_r * N_s)$$

Where T_{rs} is the sum of all pairwise distances between cluster ***r*** and cluster ***s***. N_r and N_s are the sizes of the clusters ***r*** and ***s*** respectively**.**
At each stage of hierarchical clustering, the clusters ***r*** and ***s***, for which ***D(r,s)*** is the minimum, are merged. The figure below illustrates average linkage clustering:



Average Linkage Clustering

Average Group Linkage

With this method, groups once formed are represented by their mean values for each variable, that is, their mean vector, and inter-group distance is now defined in terms of distance between two such mean vectors.

In the average group linkage method, the two clusters **r** and **s** are merged such that, after merger, the average pairwise distance within the newly formed cluster, is minimum. Suppose we label the new cluster formed by merging clusters **r** and **s**, as **t**. Then **D(r,s)** , the distance between clusters **r** and **s** is computed as

D(r,s) = Average { d(i,j) : Where observations i and j are in cluster **t**, the cluster formed by merging clusters **r** and **s** }

At each stage of hierarchical clustering, the clusters **r** and **s** , for which **D(r,s)** is minimum, are merged. In this case, those two clusters are merged such that the newly formed cluster, on average, will have minimum pairwise distances between the points in it.

Ward's Hierarchical Clustering Method

Ward (1963) proposed a clustering procedure seeking to form the partitions P_n, $P_{n-1,.....,}$ P_1 in a manner that minimizes the loss associated with each grouping, and to quantify that loss in a form that is readily interpretable. At each step in the analysis, the union of every possible cluster pair is considered and the two clusters whose fusion results in minimum increase in 'information loss' are combined. Information loss is defined by Ward in terms of an error sum-of-squares criterion, ESS.

The rationale behind Ward's proposal can be illustrated most simply by considering univariate data. Suppose for example, 10 objects have scores (2, 6, 5, 6, 2, 2, 2, 2, 0, 0, 0) on some particular variable. The loss of information that would result from treating the ten scores as one group with a mean of 2.5 is represented by ESS given by,

$$ESS_{\text{One group}} = (2 - 2.5)^2 + (6 - 2.5)^2 + \ldots\ldots + (0 - 2.5)^2 = 50.5$$

On the other hand, if the 10 objects are classified according to their scores into four sets,

{0,0,0}, {2,2,2,2}, {5}, {6,6}

The ESS can be evaluated as the sum of squares of four separate error sums of squares

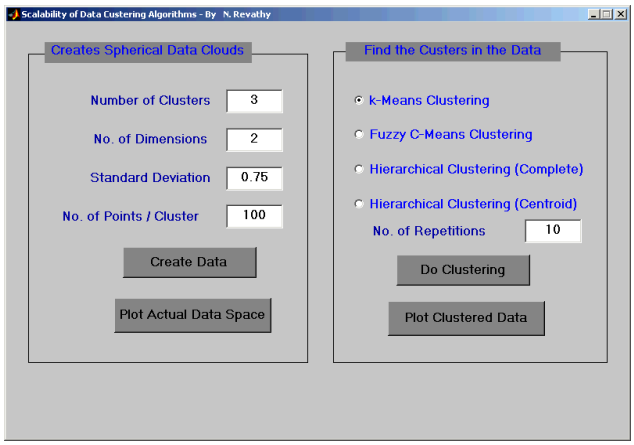
$$ESS_{\text{One group}} = ESS_{\text{group1}} + ESS_{\text{group2}} + ESS_{\text{group3}} + ESS_{\text{group4}} = 0.0$$

Thus, clustering the 10 scores into 4 clusters results in no loss of information.

THE MAIN INTERFACE DESIGNED FOR ANALYSIS

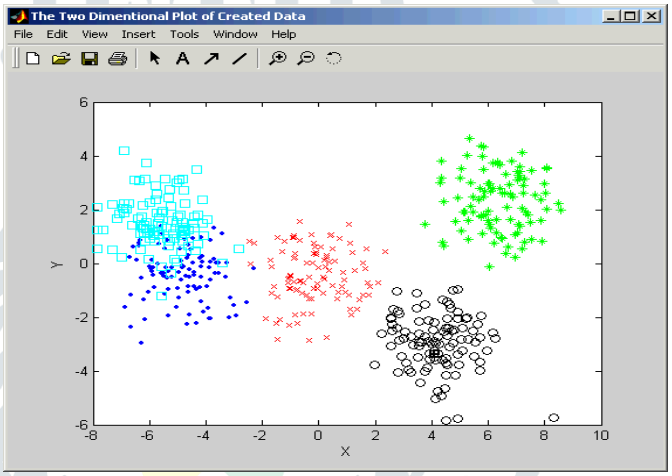
The following GUI Interface was designed to modify some input parameters during evaluating the algorithms. The Controls in the left side of the form were used to control the artificial data creation. The right side controls were used to select a algorithm for evaluation.

The Main Interface



The following graph shows a 2 dimensional plot of original artificial data clusters belongs to five classes.

The 2 D Plot of Artificial Data



Agglomerative (Hierarchical) Clustering - Complete

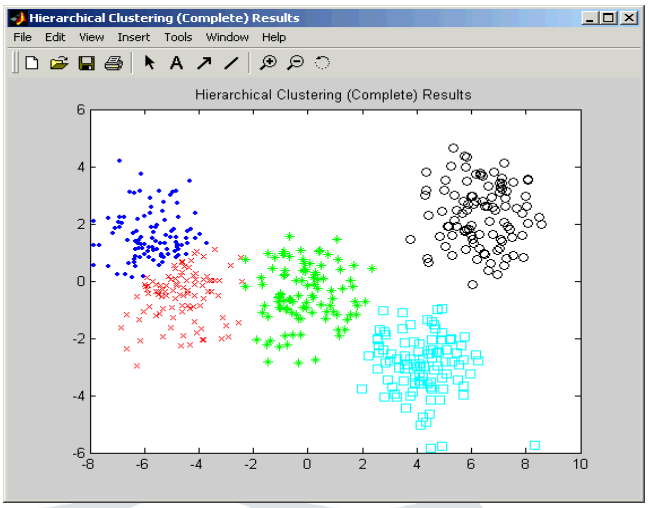
Hierarchical Clustering (Complete) Results:

Total Number of Clusters	: 5 Nos
Number of Dimension of Data	: 2
Standard Deviation (width)	: 1
Total Number Points/Cluster	: 100 Points/Cluster
Total Number Points	: 500 Points

Hubert & Arabie adjusted Rand index

Rand index of Calculated and True Classes	: 0.7837
Self Rand index of Calculated Classes	: 0.8078
Total Number of Repetitions	: 10 sec
The Time Taken for Clustering	: 0.140000 sec

The above graph shows a 2 dimensional plot of data as per the classification by Hierarchical Clustering (Complete) algorithm.



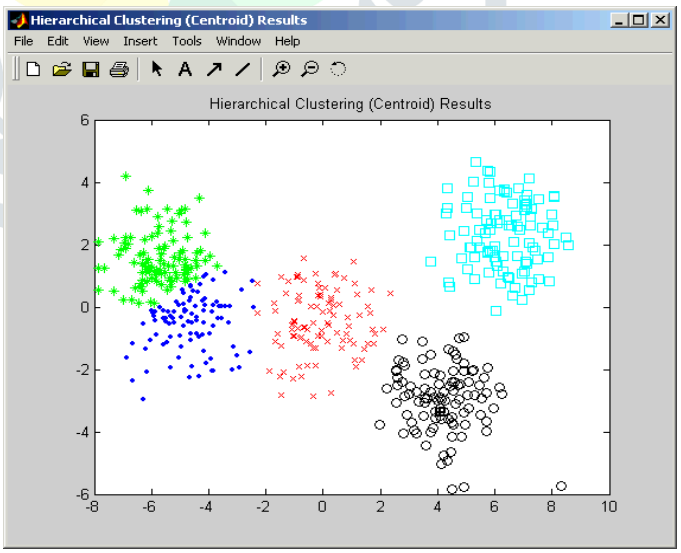
Agglomerative (Hierarchical) Clustering - Centroid

Hierarchical Clustering (Centroid) Results:

Total Number of Clusters	: 5 Nos
Number of Dimension of Data	: 2
Standard Deviation (width)	: 1
Total Number Points/Cluster	: 100 Points/Cluster
Total Number Points	: 500 Points

Hubert & Arabie adjusted Rand index

Rand index of Calculated and True Classes	: 0.8217
Self Rand index of Calculated Classes	: 0.8545
Total Number of Repetitions	: 10 sec
The Time Taken for Clustering	: 0.120000 sec



THE OVERALL RESULTS

Dimension of data	:	2
Total Clusters	:	5
Standard Deviation	:	0.75
Number of Repetitions	:	10

Overall Results

Algorithm	Total Number Points/ Cluster	Total Number Points	Rand index	Self Rand index	The Time Taken for Clustering (in Seconds)	Average Classification Time per point (in Seconds)
Hierarchical Clustering (Complete)	500 1000 1500 2000 2500 3000	2500 5000 7500 10000 12500 15000	0.692 0.784 0.779 0.873 0.760 0.804	0.990 0.822 0.833 1.000 0.813 0.927	0.791 1.432 2.964 3.625 7.601 7.441	0.000454
Hierarchical Clustering (Centroid)	500 1000 1500 2000 2500 3000	2500 5000 7500 10000 12500 15000	0.793 0.719 0.682 0.736 0.744 0.791	0.705 0.999 0.727 0.690 0.718 0.830	0.811 1.933 3.205 5.428 7.803 8.901	0.000535

During the tests, the first and second algorithms also behaved in the same manner and results were also nearly equal. Average classification time for a single data point for the four algorithms were as follows:
Hierarchical Clustering (Complete) : 0.000454
Hierarchical Clustering (Centroid) : 0.000535
Average Performance per Point : 0.000440

As per the results, the two algorithms performed almost same in terms of speed. The results beyond 15000 points were not presented since all the four algorithms behave in a peculiar manner because of the scalability issues. In some cases, most of the time the results were random beyond 15000 points.

The following Graph shows the classification time taken for each algorithm for different number of data points. This proves that if the total number of points is below 15000, the algorithms behaved normal in that particular computer in which the algorithms were implemented and evaluated. Since the scalability issues are very much depend on the hardware capabilities, the results may slightly vary from one hardware to another.

CONCLUSION AND SCOPE FOR FURTHER ENHANCEMENT

An elaborate exploration was made on classical and modern data mining algorithms. Agglomerative clustering (Complete & Centroid) algorithm was reviewed and implemented on Matlab for studying the scalability performance of this algorithm.

For the final results, the algorithm was tested with number of points between 500 to 3000 per class. For all iterations, the number of classes was five and the number of dimension was two. While testing the algorithm between 500 to3000 points per class, the test was done repeatedly. In this case, the results were acceptable below 2500 points per class. But, greater than 2500 points per class, the performance was not seemed to be linear. For greater 3000 points per class, the algorithm took very long time. This proves the inability of that algorithm for handling bulk amount of data.

The performance of agglomerative clustering (Complete and Centroid) algorithm was same in the terms of speed. From the implementation results it was obvious that the algorithm which were implemented and tested were not scalable. The performance of the algorithm was found to be accurate if the total number of data, the total number of classes in the data and the dimension of each of the points in the data were comparatively very low.

REFERENCES:

[1]. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A. "Fast Discovery of Association Rules".

[2]. Blockeel, H., De Raedt, L. "Relational Knowledge Discovery in Databases", Proceedings BENELEARN-96.

[3]. Chattratichat, J. et al, "Large Scale Data Mining: Challenges and Responses", Proceedings KDD '2010.

[4]. Gabowski, H., Lossack and Weibkopf, "Automatic Classification and Creation of Classification Systems Using Methodologies of Knowledge Discovery in Databases,"

[5]. Galal, G., Cook, D.J., Holder, L.B. "Improving Scalability in a Scientific Discovery System by xploiting Parallelism", Proceedings KDD '2011.

[6]. Holsheimer, M., Kersten, M., Mannila, H., Toivonen, H. "A Perspective on Databases and Data Mining", Proceedings KDD '95.

[7]. John, G.H., Lent, B."Sipping from the Data Firehose", Proceedings KDD '2010.

[8]. Porter, A. L., Kongthon, A., and Lu, J. C.) "Research Profiling – Improving the Literature Review: Illustrated for the Case of Data Mining of Large Datasets,"

[9]. Toivonen, H. "Discovery of frequent patterns in large data collections", PhD Thesis, 2006.

[10]. Teófilo Campos, "PCA for face recognition" Creativision research group, IME - USP - Brazil

[11]. Srikant, R., Agrawal, R. "Mining Generalized Association Rules", Proceedings VLDB '2008.

[12]. Toivonen, H. "Discovery of frequent patterns in large data collections", PhD Thesis, 2010.

[13]. N.Revathy, T.Guhan, S.Selvarajan, A Study on the Scalability of Classical Data Clustering K-Means Algorithm, International Journal of Advances in Engineering & Technology, Apr., 2017. ©IJAET ISSN: 22311963 159, Vol. 10, Issue 2, pp. 159 – 174