Privacy Preserving Density Based Clustering of Multi-Dimensional Data (DBCMD)

Submitted By

Muhammad Sajid Qureshi

366-FBAS/MSCS/F07

Supervised By

Muhammad Imran Saeed

Department of Computer Science & Software Engineering

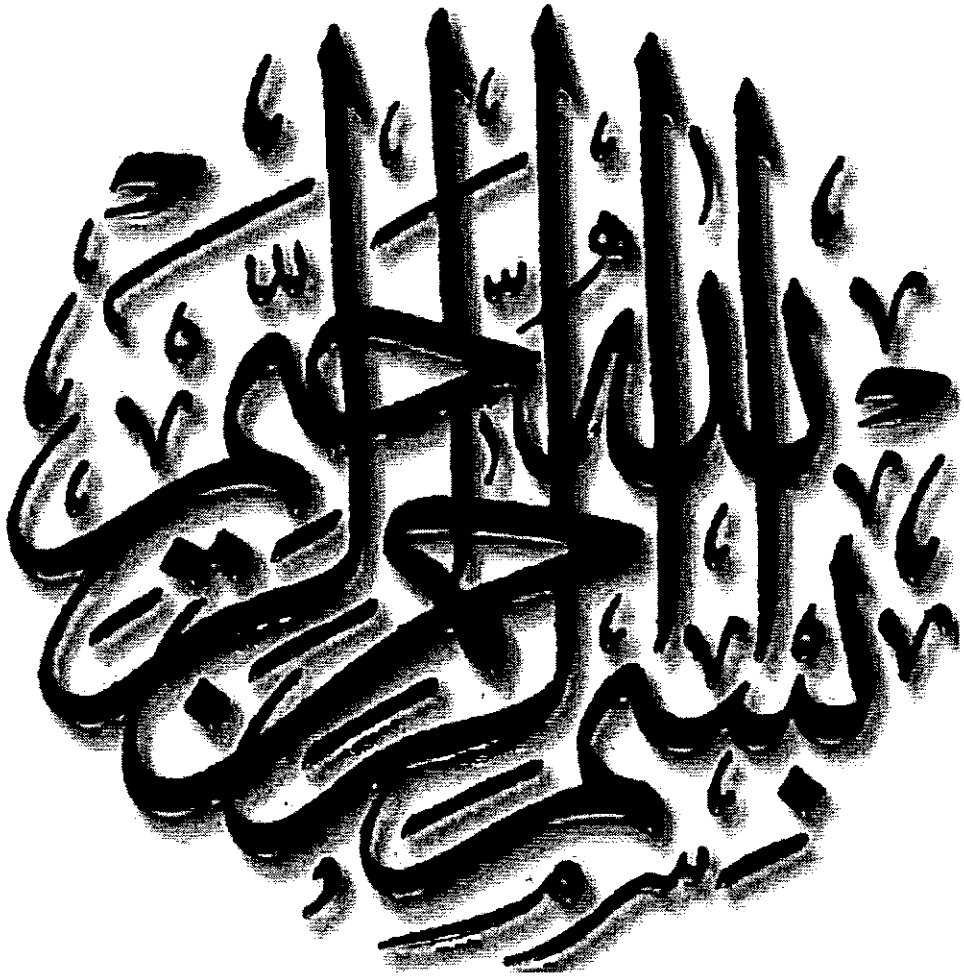
Faculty of Basic & Applied Sciences

International Islamic University Islamabad

2012

MS
004
QUP

1. Data processor
2. Computer Science



In The Name Of

ALLAH

The Most Beneficent, the Most Merciful

This

Piece of work is

Dedicated to the People

Who

Contributed in My

Education

This Dissertation is

Submitted to the Department of

Computer Science & Software Engineering,

International Islamic University, Islamabad

As a partial fulfillment of the requirements for

Award of the degree of Masters in

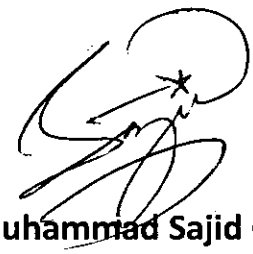
Computer Science

MS (CS)

Declaration

I hereby solemnly declare that this research work "***Privacy Preserving Density Based Clustering of Multi-Dimensional Data***" neither as a whole nor as a part has been copied out from any source. I have done this research, with the accompanied report, entirely on the basis of my personal efforts, under the proficient guidance of my research supervisor Mr. Muhammad Imran Saeed.

It is to be stated further that this research thesis and its accompanied software application code has been declared "Plagiarism Free" by the HEC (Higher Education Commission) recommended web portal www.turnitin.com. According to the *Turnitin Originality Report* the Overall Similarity Index was only **14%** which lies within the acceptable range.



Muhammad Sajid Qureshi
MS (Computer Science)
366-FBAS/MSCS/F07

Acknowledgement

I simply bow my head before Allah Almighty for giving me faith in my abilities and enabling me to accomplish this research work. He showered on me His unremitting blessings throughout my life.

Parent's prayers remained as an asset for me throughout my career. Their unparallel self-sacrifices always inspired me to put my best efforts in every challenge of my life. The humble accomplishment of this thesis would not have been possible without the contribution of many individuals, to whom I must express my appreciation and gratitude.

I am especially grateful to Mr. Muhammad Imran Saeed – my research supervisor for his guidance, encouragement and support in all phases of the research activity. I am also thankful to Mr. Asim Munir – for his extended cooperation and support during the research work. He remained a source of inspiration for me.

I must be thankful to my colleagues who encouraged me in the difficult time and extend their helping hand toward me; especially Abdul Nasir and Mr. Imran Babar. They assisted me in preparation of the end user application.

In the nutshell, it is the blessings of Almighty Allah, parent's prayers, guidance of my supervisor, and the encouragement of my colleagues which enable me to complete this piece of work successfully.

Muhammad Sajid Qureshi

Abstract

This dissertation addresses a sensitive cotemporary problem of the *Data Mining* process—*the need to preserve privacy of target data during the data mining process*. It proposes and demonstrates an effective privacy preserving technique for Density Based Clustering of Multi-Dimensional Data. In fact, it focuses on addition of the data privacy preservation feature, to the well-known density base algorithm – Density Based Spatial Clustering of Applications with Noise (DBSCAN). The value added version of DBSCAN employs a combination of data encryption techniques to preserve privacy of the data.

The employed privacy preservation technique ensures retention of *computational value* of the data while preserving data privacy by using the linear data transformation method to encrypt record of an individual record. The method maps original value of an attribute of a record to a predefined mapped value from the mapping table, according to magnitude of original value of the attribute. Such proportional mapping or transformation significantly preserves privacy of a record without losing the computational value of data.

International Islamic University Islamabad

12th June-2012

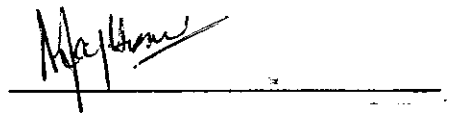
Final Approval

It is certified that we have read the thesis titled **"Privacy Preserving Density Based Clustering of Muiti-Dimensional Data"** submitted by **Mr. Muhammad Sajid Qureshi**. It is our judgment that this thesis is of sufficient standard to warrant its acceptance by the International Islamic University, Islamabad for the **MS Degree** in **Computer Science**.

Committee

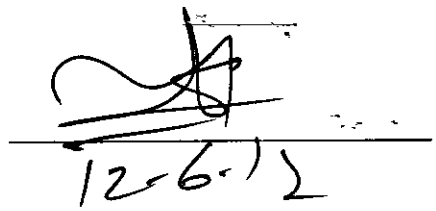
External Examiner

Dr. Syed Afaq Husain
Department of Computing,
Riphah International University
Islamabad.



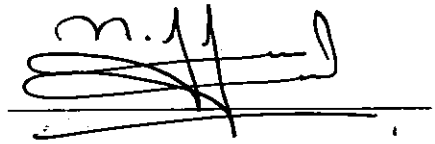
Internal Examiner

Professor Dr. Muhammad Sher
Head of Department of Computer Science & Software Engineering
International Islamic University, Islamabad.



Supervisor

Mr. Muhammad Imran Saeed
Assistant Professor,
Department Of Computer Science & Software Engineering
International Islamic University, Islamabad.



List of Figures

List of Tables

Contents

3. Proposed solution

4. Implementation

5. Conclusions and Future Work

Chapter 1

Introduction

1. Introduction

In the last few decades, tremendous progress has been observed in the data processing capability, data storage capacity, computers inter-networking and the electronic data management methods. This progress is resulting into an unprecedented amount of digitization of data and information. The digital data processing has covered a long distance, passing from the *File Processing Systems* to the *Database Management Systems (DBMS)*; it has now reached to the Data *Warehousing* and the *Data Mining*. This over-whelming growth of the computing technology and emergence of electronic data management methods has persuaded some social scientists to rightly call the present age as the "Information Age".

The Information age has enabled many governments, enterprises and the organizations to gather large volumes of data. Data Warehouses and the Data Marts are becoming a necessity for them. However, the usefulness of those large repositories of data is negligible if the "meaningful information" or "knowledge" cannot be extracted from them. This need gave birth to the Data Mining (or Automated Extraction of useful knowledge from huge amounts of data). With every passing year, more and more organizations, companies and institutions that deal with large databases are adopting the Data Mining.

1.1. Data Mining and its Significance

"Data Mining is a process of extracting valid, previously unknown, comprehensible and actionable information from large databases; and using it to make crucial business decisions." **[1]**. Data Mining is also known as *Knowledge Discovery* from the Database. Alternatively:

"Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in innovative ways that are both understandable and useful to the data owner." **[2]**

So, the Data Mining analyzes data from different perspectives and extracts hidden patterns. These analyzed facts and patterns are summarized into useful information. Data Mining is among the emerging fields in the computing technology and connects the three major technical areas-- Databases, Artificial Intelligence and the Statistics. The information age has enabled many organizations to gather large volumes of data. However, the usefulness of this data is negligible if "meaningful information" or "knowledge" cannot be extracted from it. Data mining, otherwise known as knowledge discovery, attempts to answer this need. **[3]**

The nature of Data Mining is entirely interdisciplinary; it attracts experts, researchers and interested audiences from different fields such as Computer Science, Database Systems, Artificial

Intelligence, Biology, Agriculture, Business, Mathematics, and Statistics. They all have a common objective of extracting useful but hidden knowledge that can aid in decision-making processes and which would have remained unknown otherwise.

The advances in information processing technology and the storage capacity have established the Data Mining as a widely accepted technique in the decision making process. Many organizations, dealing with large databases, are highly dependent on Data Mining in their functioning.

Data Mining has multiple definitions such as "Non-trivial extraction of implicit, previously unknown and potentially useful information from data" or "Exploration & analysis, by automatic or semi-automatic means, of large quantities of data in order to discover meaningful patterns" etc. the following diagrams facilitates understanding of the data mining process.

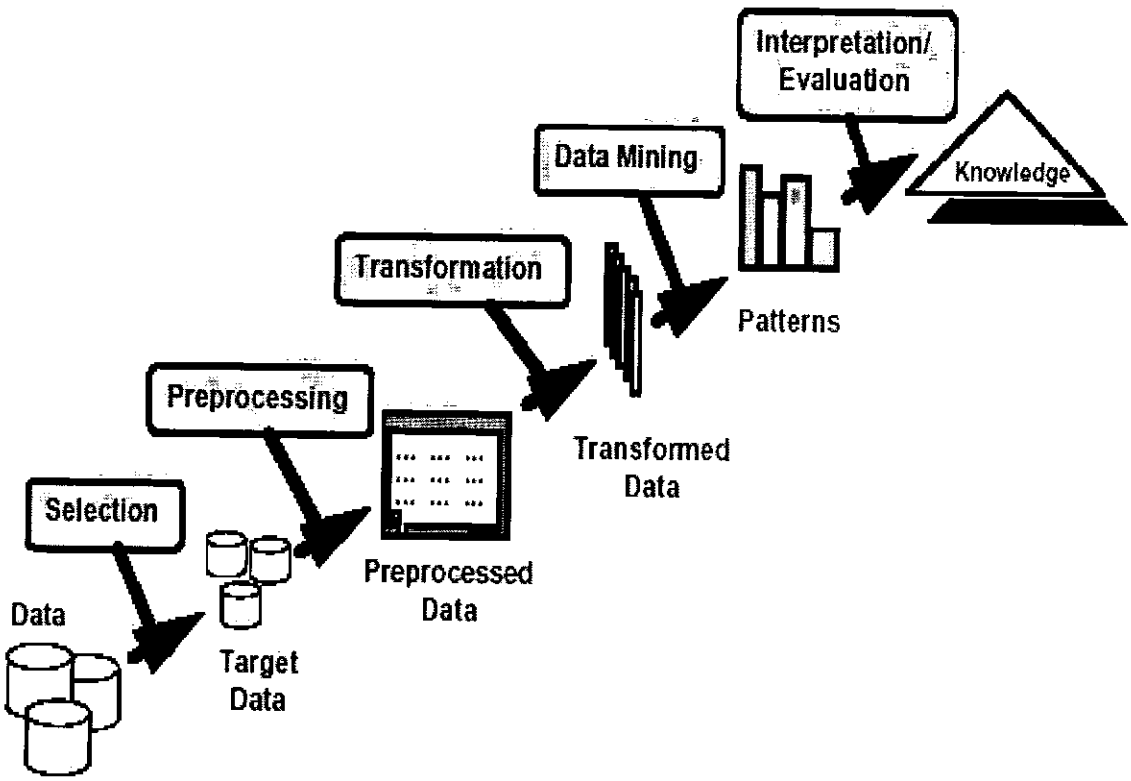


Figure 1.1: Illustration of Different Phases of the Data Mining Process

1.2. Applications of the Data Mining

Data Mining has great importance in today's highly competitive business environment. It is based on mathematical algorithm and analytical skills to drive the desired results from the huge database collection. Data Mining is largely being used in several sectors; some of them are mentioned below:

- Data mining has attracted a great deal of attention in the information industry and in society as a whole in recent years, due to the wide availability of huge amounts of data and the imminent need for turning such data into useful information and knowledge.
- Data mining is available in various forms like text mining, web mining, audio & video data mining, pictorial data mining, relational databases, and social networks data mining.
- Its applications are such as understanding consumer research marketing, product analysis, demand and supply analysis, e-commerce, investment trend in stocks & real estates, telecommunications and so on.
- Data mining applications are widely used in direct marketing, health industry, e-commerce, customer relationship management (CRM), telecommunication industry and financial sector.
- The information and knowledge gained can be used for applications ranging from market analysis, fraud detection, and customer retention, to production control and science exploration.
- Data Mining has been used in different context but is being commonly used for business and organizational needs for analytical purposes.
- A new concept of Business Intelligence (BI) data mining has evolved now, which is widely used by leading corporate houses to stay ahead of their competitors. Business Intelligence (BI) can help in providing latest information and used for competition analysis, market research, economical trends, consume behavior, industry research, geographical information analysis and so on. Business Intelligence Data Mining helps in decision-making.

Data mining, however, is a difficult and complicated process so requires lots of time and patience in collecting desired data due to complexity and of the databases. This could also be possible that you need to look for help from outsourcing companies. The outsourcing companies are specialized in extracting or mining the data, filtering it and then keeping them in order for analysis.

1.3. The Privacy Preserving Data Mining (PPDM)

Use of the Data mining is resulting in more effective decision-making, providing better services, and achieving greater profit in the business. For these purposes the governmental institutions, enterprises and organizations collect huge amount of data on which they apply the Data Mining techniques. For example, business organizations collect data about the trends and habits of their consumers for marketing purposes and to improve their business strategies. Similarly medical organizations collect medical records to provide better treatment to their patients and for the medical research.

Many data mining applications deal with large data sets that contain private information that must be protected. In last few years, along with the dramatic increase in digital data, concerns about the privacy of personal information have also emerged globally. Data mining, with its promise to efficiently discover valuable but hidden information from large databases, is particularly vulnerable to misuse. The challenge for the Database and Data Mining community is to design information systems that protect privacy and ownership of the individual data without losing the effectiveness of the available data

While all of these applications of data mining can benefit humans and save lives, there is also a negative side to this technology, since it could be a threat to the privacy of individuals and civil liberties. Therefore the future database systems must include responsibility of privacy of the data they manage, as a founding principle. Similarly the other data processing applications like Online Analytical Processors (OLAP) or the Data Mining applications must be sensitive to the issue of the Data Privacy. A fruitful direction for future data mining research will be the development of techniques that incorporate privacy concerns; in other words we are in need of the Privacy Preserving Data Mining (PPDM).

In context of the Knowledge Discovery Databases (KDDs) the Data privacy issue is known as the *Inference Problem* **[4, 5].** In this process the user of a Database poses authorized queries and deduces unauthorized information from the legitimate responses to the queries. This problem has been discussed quite a lot over the past two and half decades in multiple communities such as the Database community, the Statistical Disclosure Control community and the Cryptography community. However, data mining makes this problem worse. Users now have sophisticated tools that they can use to get data and deduce patterns that could be sensitive. That is, data mining tools make the inference problem quite dangerous.

1.4. Major Functionalities of Data Mining

In general Data mining tasks are classified into two categories:

- *Descriptive Data Mining:* This method is used to characterize the general properties of data in the database.
- *Predictive Data Mining:* This method is used to perform inference on the current data in order to make predictions about the future data (future trends).

Here is a list of the Data mining functionalities that discover different kinds of patterns. Each of the function requires in-depth study and analysis for its due understanding, but here we are interested in the Data Clustering that is to be in the research work.

a) Characterization and Discrimination of the Data sets
b) Mining Frequent Patterns, like Associations, Subsequences and the Correlations
c) Classification and Prediction
d) ***Data Clustering***
e) Outlier Detection and Analysis
f) Evolution Analysis

1.5. Major Issues of Data Mining

The Concept of Data Mining surfaced during 1980-90. After that various techniques and algorithms were proposed and implemented in the last two decades. Currently Data Mining experts and researchers are dealing with the various issues related to the Data Mining regarding Data Mining methodology, user interaction, performance and diverse data types. Here is a brief list of the major issues of Data Mining:

Mining methodology and user interaction issues

a. Mining different kinds of knowledge in databases
b. Data mining query languages and ad hoc data mining
c. *Preserving the Privacy of Data in the Data Mining Process*
d. Presentation and visualization of data mining results
e. Interactive mining of knowledge at multiple levels of abstraction
f. Incorporation of background knowledge
g. Handling noisy or incomplete data
h. Pattern evaluation—the interestingness problem

Performance issues

i. Efficiency and scalability of data mining algorithms.

j. Parallel, distributed, and incremental mining algorithms.

k. Issues relating to the diversity of database types.

l. Handling of relational and complex types of data.

m. Mining the Heterogeneous Databases and Global Information Systems.

1.6. Density Based Clustering and the DBSCAN

Density-based clustering algorithms are designed to discover arbitrary-shaped clusters. In this approach, a cluster is regarded as a region in which the density of data objects exceeds a threshold or limit. DBSCAN and OPTICS are two examples of typical density based algorithms.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a data clustering algorithm proposed by Martin Ester, Hans-Peter etl. in 1996. It is a density based clustering algorithm because it finds a number of clusters starting from the estimated density distribution of corresponding nodes. DBSCAN is one of the most common clustering algorithms and also most cited in scientific literature.[6]

A density-based cluster is a set of density-connected objects that is maximal with respect to density-reachability. Every object not contained in any cluster is considered to be noise.

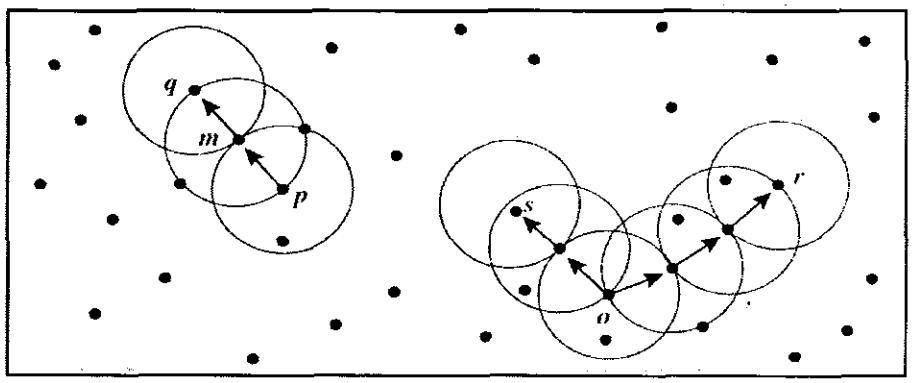


Figure 1.2: Density-Reachability and Density Connectivity in density-based clustering

In the above figure, of the labeled points:

- ***m***, ***p***, ***o***, and ***r*** are core objects because each is in an é-neighborhood containing at least three points. ***q*** is directly density-reachable from ***m***. ***m*** is directly density-reachable from ***p*** and vice versa.

- ***q*** is (indirectly) density-reachable from ***p*** because ***q*** is directly density-reachable from ***m*** and ***m*** is directly density-reachable from ***p***. However, ***p*** is not density-reachable from ***q*** because ***q*** is not a core object.
- Similarly, ***r*** and ***s*** are density-reachable from ***o***, and ***o*** is density-reachable from ***r***.
- ***o***, ***r***, and ***s*** are all density-connected [7].

DBSCAN (Density Based Spatial Clustering of Applications with Noise) is one of them, with ability to detect arbitrary shaped clusters. DBSCAN algorithm grows regions with sufficiently high density into clusters and discovers clusters of arbitrary shape in spatial databases having noise. Density is defined as a minimum number of points within a certain distance (ε) between each other points of a cluster. It takes input parameters ε (the minimum radius) and *MinPts* (minimum number of points in any cluster at a distance ε).

DBSCAN searches for clusters by checking the ε-neighborhood of each point in the database. If the ε-neighborhood of a point p contains more than *MinPts*, a new cluster with **p** as a core object is created. DBSCAN then iteratively collects directly density-reachable objects from these core objects, which may involve the merge of a few density-reachable clusters. The process terminates when no new point can be added to any cluster. All the non-core objects which are not in the neighborhood of any of the core objects are labeled as noise.

DBSCAN doesn't need the number of final clusters to be given in advance. There are several clustering algorithms are available. They are applicable to specific type of data, but DBSCAN is applicable for all types of data and the clusters obtained by DBSCAN are similar to natural clusters.

Advantages

- DBSCAN can find clusters with arbitrarily shape unlike the many other clustering algorithms those can find clusters with circular shape only. It can even find clusters completely surrounded by, but not connected to a different cluster.
- DBSCAN does not require you to know the number of clusters in the data before the clustering process, as various other algorithms require it as an input parameter K-means.
- DBSCAN requires just two parameters and is mostly insensitive to the ordering of the points in the database. Only points sitting on the edge of two different clusters might swap cluster membership if the ordering of the points is changed, and the cluster assignment is unique only up to isomorphism.
- DBSCAN has a notion of noise in the dataset; therefore it can detect the outlier data items effectively.

Disadvantages

- DBSCAN does not operate well on the hierarchical data sets (data sets with varying densities).
- The DBSCAN can give as good result as its distance measure is good in its function. Commonly the Euclidean distance measure is used with the DBSCAN, but for high-dimensional data, this distance metric become ineffective.

Complexity

DBSCAN requires to test each point of the database, possibly multiple times (e.g., to check candidacy for different clusters). In practice, however, the time complexity is dependent on the number of searches for the neighbors to the Core object. It performs exactly one such search for each point in the dataset, in such case the complexity may be **O(log n)** where ***n*** is number of the points in the dataset. (This requires use of an indexing structure to executes such a neighborhood search)

1.7. Privacy Preserving Data Clustering (PPDC)

Being a dynamic field of research in data mining, the data clustering is done through various algorithms. These algorithms had been developed and implemented in last three decades so they are time tested. Normally these algorithms are categorized in various groups. Like the Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Model-Based Methods, Methods For High-Dimensional Data (including frequent pattern–based methods), and Constraint Based Methods.

As for as privacy preserving data clustering is concerned, we can define it as a clustering process which employs an algorithm that minimizes secondary use of data or the breach of data privacy during the clustering process. As the issue of privacy of an individual's record became serious in the last two decades, it affected the data clustering domain also. The data clustering experts and scholars have to pay attention to the issue of privacy. Therefore the literature of last ten to fifteen years contains various proposed algorithms along with the claim of privacy preservation of the data to be clustered.

As it is clear from definition of the data clustering that it requires construction of the *Data Dissimilarity Matrix* by use of a distant measure (i.e Manhattan Distance, Euclidean Distance Measure etc.). The distance measure uses the attributes or properties of an individual data item to compare it with the remaining items. Such computation demands preservation of following features of the records in the dataset:

- ***Complete access to the database:*** The Data Clustering requires complete database scan to compare attributes or properties of an individual data item to compare it with the remaining items, as this comparison indicates the similarity (or dissimilarity) of the data items. So any privacy preserving technique should not restrict the access to the all items / records in the database. In case of restricted or partial access to the target database, the resultant clusters will not include all data items/ records so they can not be generalized for whole database with the desired accuracy and reliability.

- ***Preserving the attributes of transaction:*** In the data clustering the distance or difference between among records/items is based on their features or attributes, so any privacy preserving technique should not disturb the original features properties or attribute of a transaction or record. Such a distortion reduces the computational value of record significantly. Similarly the perturbation of the attributes that distorts or eliminates the inherent difference /distance among the transaction or records is also harmful for the accurate and reliable data clustering.

Keeping in view the above mentioned precautionary measures, it is worth to mention that the *data clustering requires complete access to the database and demands preservation of the attributes of a transaction.* Hence we have to employ a privacy preserving data mining technique which ensures minimum or preferably ideally no distortion of the attributes of an individual record. The privacy preserving method also should not reduce the access to the complete database to be clustered.

1.8. Research Problem

In last few decades, the rapid advancement in information processing technology and availability of affordable large storage capacity, are paving ways for various organizations to adopt the Data Mining technique in their decision making process. More and more data owners are relying on Data Mining in their day to day business function. Data mining process is resulting in more effective decision-making and earning larger revenues through their better quality of services.

Many Data Mining applications deal with large data sets that contain private information that must be protected. Data mining, with its promise to efficiently discover valuable but hidden information from large databases, is particularly vulnerable to misuse. Typically, the data collection is done with the consent of the owners of the Data, and the collector provides some assurance that the privacy of individual data will be protected. Additionally, some organizations sell the collected data to other organizations, which use these data for their own purposes. Thus the data get exposed to a number of

parties including collectors, owners, users and the miners. The challenge for the Database and Data Mining community is to design information systems that protect the privacy and ownership of individual data without losing the effectiveness of the available data.

Data mining is too domain specific and requires considerable customization of the existing algorithms and the privacy preserving techniques according to the level of required privacy. Selection of the data clustering algorithm depends on type of the target dataset available and on the objectives of the clustering process. Therefore the data mining experts need to develop specific privacy preserving algorithms fulfill wide ranged demands of the computing industry.

It is evident from the current market trends and the existing data privacy concerns that the data mining has potential to capture capturing its due share from the market. Nevertheless, without privacy assurance, it would face a decline instead of capturing its due share in the software industry. It has become need of the hour to focus on the development of such Data Mining techniques that mine database in a privacy preserving way, without limiting access to the crucial data.

Chapter 2

Literature Survey

2. Literature Survey

Privacy preserving Data Mining is not a new topic of research for the young researchers. Various international conferences has been held on Data Mining and its related issues in the last few years; Similarly hundreds of research papers and archives of proceedings of the conferences, and dozens of implemented privacy preserving techniques are available for the novice researchers. ACM (Association of Computing Machinery) and IEEE web portals along with the famous search engine--Google provide a sublime collection of related material on this topic. It is necessary for the scholars and novice researchers to consult the existing material before starting their work. The following section consists of review of the existing privacy preserving techniques.

In the last few decades, tremendous progress has been observed in the Data Processing capability, Data Storage capacity, Computers Inter-Networking and the Electronic Data Management methods. This progress is resulting into an unprecedented amount of digitization of information. The digital data processing has covered a long distance, passing from the File Processing Systems to the Database Management Systems (DBMS); it has now reached to the Data Warehousing and the Data Mining. This over-whelming growth of the computing technology and emergence of electronic data management methods has persuaded some social scientists to rightly call the present age as the "Information Age". The overwhelming growth is resulting accumulation of huge repositories of data.

The Information age has enabled many governments, enterprises and the organizations to gather large volumes of data. Data Warehouses and the Data Marts are becoming a necessity for them. However, the usefulness of this data is negligible if the "meaningful information" or "knowledge" cannot be extracted from it. This need gave birth to the Data Mining (or Automated Extraction of useful knowledge from huge amounts of data). With every passing year more and more organizations, companies and institutions that deal with large databases are adopting the Data Mining.

Many data mining applications deal with large data sets that contain private information that must be protected. In previous few years, along with the dramatic increase in digital data, concerns about the privacy of personal information have also emerged globally. Data mining, with its promise to efficiently discover valuable but hidden information from large databases, is particularly vulnerable to misuse. The challenge for the Database and Data Mining community is to design information systems that protect the privacy and ownership of individual data without losing the effectiveness of the available data.

2.1. Privacy Preserving Data Clustering – Existing techniques

Clustering is a process of grouping or partitioning a set of physical or abstract objects or items into classes of similar objects. The objects within a Cluster significantly resemble each other (Intra-Cluster Similarity); while the objects of a cluster are significantly differ from the objects in other clusters (Inter-Clusters Dissimilarity). Usually the partitioning of object is done by measuring Dissimilarity (or Similarity) using a distance measure like Euclidean, Manhattan, Hamming and Minkowski distances.

The quality of a Clustering technique can be assessed based on a measure of dissimilarity of objects, which can be computed for various types of data, including interval-scaled, binary, categorical, ordinal, and ratio-scaled variables, or combinations of these variable types.

Clustering is also called *Data Segmentation* or an *Unsupervised Learning* as it does not rely on predefined classes and class-labeled training examples. For this reason, clustering is a form of learning by observation, rather than learning by examples. Unsupervised learning deals with designing classifiers from a set of unlabeled samples. A common approach for unsupervised learning is to first cluster or group unlabeled samples into sets of samples that are "similar" to each other. Once the clusters have been constructed, we can design classifiers for each cluster using standard techniques such as Decision-tree learning

Clustering is a dynamic field of research in data mining. Various clustering algorithms have been developed and implemented. These can be categorized into Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Model-Based Methods, Methods For High-Dimensional Data (including frequent pattern–based methods), and Constraint Based Methods.

The advantages and implications of the Data Clustering render it among the primary functionalities of the Data Mining. The primary task in Data Mining / Clustering is development of the models about aggregated data. If the data under processing is sensitive and requires high privacy then the Data Mining researchers have to adopt the algorithms and techniques that enable them to process the database accurately without breaching privacy of the individual records. In past, various techniques have been proposed and demonstrated in order to preserve privacy of the data. Data mining is too domain specific and requires considerable customization of algorithm and the privacy preserving technique according to the level of required privacy. The choice of clustering algorithm depends both on the type of data available and on the particular purpose of the application. Here is a brief description of the existing privacy preserving clustering techniques.

2.2. Privacy Preserving Clustering by Data Randomization (Perturbation)

The Data Randomization is a technique for privacy-preserving data mining in which randomized noise is added to the attribute values of the sensitive record, in order to preserve their privacy. If the noise addition is sufficiently lager then, in most cases, the individual records cannot be recovered from the sanitized database, although the aggregate distribution can be recovered. These aggregate distributions can be used in the data mining purposes. Data perturbation technique became popular due to its simplicity in terms of implementation. To perturb the data through Randomization, following methods are used.

- **Additive Perturbation**

In this case, a randomized noise is added to the values of attributes of the records in the dataset. For example in a national database containing personal profiles of the citizen, the data owner can decide to perturb the actual age of an individual by adding 'five' years in each age value of the even number records, while the 'five' years are deducted from the actual age of the odd number records. The overall data distributions can be recovered from the randomized records. Similarly the monthly income of an individual can be hidden by adding or deducting a predefined amount from each record.

Such additive perturbation makes it difficult for the adversary user to guess the original record of an individual, although some researchers argue that it is not sufficient since the released data contains other information which, when linked with other datasets, can identify or narrow down the individuals or entities. In addition to the identity identification problem, attribute disclosure occurs when something about an individual is learnt from the publically available data. Attribute disclosure can help an adversary user to infer some characteristics of an individual more accurately because of the data release. Attributes whose disclosure needs to be protected in the strictest sense are denoted to be sensitive and their privacy must be protected.

- **Multiplicative Perturbation:**

Although the data randomization to preserve data privacy is simple to implement but careless noise addition may introduce biases to the statistical parameters of the dataset, including means and variances. To avoid introduction of such biases and to preserve the statistical properties of the sensitive data, some researchers propose to use a data transformation technique named Rotation-Based Transformation (RBT) **[8]**. This method distorts only confidential numerical attributes and preserves the statistical properties of the data. In this case, the random projection or random

rotation techniques are used in order to perturb the data records. In this case, the random projection or random rotation techniques are used in order to perturb the records.

Advantages:

a) Adding a sufficiently large amount of noise makes it considerably difficult for the adversary to discover the original attribute values of a record.

b) It can be used on a selected set of one (or more) variables, without disturbing the responses for non-sensitive and non-identifying fields.

c) The randomization method is relatively simple to implement and does not require knowledge of the distribution of other records in the data.

Disadvantages:

a) The key weakness of the randomization framework is that it does not consider the possibility that publicly available records and the background knowledge can be used to identify the identity of the owners of that record.

b) It requires sufficiently large amount of noise addition to preserve the privacy of attributes of records, but this reduces the analytical value of the attributes; so privacy and accuracy become inversely proportional.

c) Usually, rare and unique responses are used to identify the respondents; Since Data Randomization treats all records equally irrespective of their sensitivity, so the outliers and unique records are susceptible to adversarial attacks even after the randomization. The randomization may be *Additive* or *Multiplicative*.

This technique has been employed by researcher like Noirin Plunkett, Stanley R. M. Oliveira and Osmar R. Zaiane **[9, 10].**

2.3. Privacy Preserving Clustering by K-Anonymity technique

Sanitization of the database is among the popular techniques to preserve the data privacy; but various attributers in the record or transaction are often considered as pseudo-identifiers which can be used in conjunction with other publicly available records, in order to uniquely identify the desired records. For example, if the unique identifiers (such as Employee ID, National / Social Security Number) from

records are removed, attributes such as the birth data, place of domicile or the zip-code can be used in order to uniquely identify the identities of the underlying records. These attributes are called quasi-identifiers or pseudo-identifiers. The can be potentially used to identify individuals by linking these attributes to external data sets.

To counter such attributes linking attacks using the pseudo-identifiers Samarati and Sweeney (2002) proposed a definition of privacy called K-Anonymity. The K-Anonymity model was developed because of the possibility of indirect identification of records from public databases. The idea in K-anonymity is to reduce the granularity of representation of other records. A table satisfies K-Anonymity if every record in the table is indistinguishable from at least k-1 other records with respect to every set of quasi-identifier attributes; such a table is called a k-anonymous table. Hence, for every combination of values of the quasi identifiers in the k-anonymous table, there are at least k records that share those values. This ensures that individual cannot be uniquely identified by the attribute-linking attacks.

To do this job, the k-anonymity model used technique such as Generalization and Suppression. This granularity is reduced sufficiently that any given record maps onto at least k other records in the data. Clearly such methods reduce the risk of identification with the use of public records, while reducing the accuracy of applications on the transformed data.

Attribute Suppression

In this method value of the sensitive or restricted attribute is removed completely. For example, the unique identifiers (such as Employee ID, National / Social Security Number) are removed from the records. However, this first sanitization still does not ensure the privacy of individuals in the data. A recent study estimated that 87% of the population of the United States can be uniquely identified using the seemingly innocuous attributes of gender, date of birth, and 5-digit zip code [Sweeney 2000].

Attribute Generalization

In this method the attribute values are generalized to a range in order to reduce the granularity of representation. For example, the date of birth could be generalized to a range such as year of birth; similarly the exact salaries can be generalized to salary range like a value "5 to 10 thousand" can be substituted in place of the salaries like 5000, 7000 or 9000 to reduce the risk of identification.

Most of the inference channels in a multilevel database are created by combining the Meta Data (e.g. database constraints) with the data to obtain the hidden sensitive information. Similarly external

knowledge and knowledge of the domain play a significant role in inferring the sensitive information of database. K-Anonymity technique is susceptible to various kinds of attacks **[11]** especially when background (Domain) knowledge is available to the attacker. Some kinds of such attacks are as follows:

- **Homogeneity Attack**

In this attack, all the values for a sensitive attribute within a group of k records have a little diversity among them. Therefore, even though the data is k-anonymized, the value of the sensitive attribute for that group of k records can be predicted exactly.

- **Background Knowledge Attack**

In this attack, the adversary can use a set of attributes (like gender, date of birth, and zip code) that can be associated with the external data to uniquely identify individuals in the database.

Advantages of K-Anonymity:

a) K-Anonymity method is more secure than the simple data perturbation methods as it increases the anonymity of a sensitive record up to K-levels.

b) It ensures to discover the interesting information and patterns from the target database, with a reliable accuracy.

Disadvantages:

a) K-Anonymity technique is susceptible to various kinds of attacks like the Background Knowledge Attack and Homogeneity Attack.

b) It loses its privacy level significantly when the sensitive values within a group are homogeneous. Therefore in some cases it becomes easy for the adversary to determine the individual records.

The k-anonymity model hides the value of individual records by ensuring that each record of a relation (table) is identical to at least k–1 other records with respect to a set of privacy-related attributes. These attributes are called quasi-identifiers, and can be potentially used to identify individuals by linking these attributes to external data sets. Meng-Cheng Wei claims to implement this technique successfully to Cluster the data in a privacy sensitive way **[12]**. He describes two levels of anonymization

- *Attribute Level*: In it the anonymization is achieved via generalization at the attribute level, i.e., if two records contain the same value at a quasi-identifier, they will be generalized to the same value at the quasi-identifier as well.

- *Cell Level*: In it the anonymization is achieved via generalization at the cell level. In this generalization, two cells with same value could be generalized to different values

Because anonymization via generalization at the Cell-level generates data that contains different generalization levels within a column, utilizing such data becomes more complicated than utilizing the data generated via generalization at the attribute level. However, generalization at the Cell-level causes less information loss than generalization at the attribute level. Hence, as far as data quality is concerned, generalization at the cell level seems to generate better data than generalization at the attribute level.

Anonymization via generalization at the cell level can proceed in two steps. First, all records are partitioned into several groups such that each group contains at least k records. Second, the records in each group are generalized such that their values at each quasi-identifier are identical. To minimize the information loss incurred by the second step, the first step should place similar records (with respect to the quasi-identifiers) in the same group.

2.4. Privacy Preserving Clustering by L-Diversity Technique

Many organizations are increasingly publishing the micro data, that is, tables that contain non aggregated information about individuals. These tables can include medical histories, voter registration, Census data, Customer's data etc. These data are a valuable resource for the medical research, allocation of the public funds and trend analysis. However, if an individual's record can be uniquely identified in the dataset, then his personal information (such as the financial status or the medical fitness level etc.) would be disclosed; such a privacy breach is undesirable and in some cases unacceptable. Processing such sensitive data without revealing sensitive information about an individual is an important problem in the computing community.

In the recent years, a new definition of privacy, called k-anonymity has gained popularity. In the k-annonymized dataset, each record is indistinguishable from at least k-1 records with respect to certain identifying attributes known as the quasi-identifiers. It is a set of attributes like gender, date of birth, zip code etc. that can be linked with the publically available data to uniquely identify individuals in the dataset.

Ashwin Machanavajjhala et al. claim by using two simple attacks that a k-anonymity privacy preserving technique has some delicate but swear privacy problems. First, an adversary database user can discover the values of sensitive attributes (quasi-identifiers) when there is little diversity in those sensitive attributes. Similarly k-anonymity does not protect against attacks based on background knowledge. We need a stronger definition of privacy that takes into account diversity and background knowledge.

The authors provide a detailed analysis of above mentioned tow attacks. They also propose a powerful privacy preserving technique called L-diversity that can defend against such attack. In addition to building a formal foundation for L-diversity, they show in an experimental evaluation that L-diversity is practical and can be implemented efficiently.

K-Anonymity can create groups that leak information due to lack of diversity in the sensitive attributes. The L-Diversity model **[13]** was designed to handle some weaknesses in the k-anonymity model since protecting identities to the level of k-individuals is not the same as protecting the corresponding sensitive values, especially when there is homogeneity of sensitive values within a group. This suggests that, in addition to k-anonymity, the sanitized table should also ensure *intra-group diversity*, that is, all tuples that share the same values of their quasi-identifiers should have diverse values for their sensitive attributes after the data sanitization

L-Diversity provides privacy even when the data publisher does not know what kind of knowledge the adversary possesses. The main idea behind L-diversity is the requirement that the values of the sensitive attributes are well represented in each group. To achieve the desired security level L-Diversity relies on the following three assumptions:

- Tuples or individual records with similar but non-sensitive attributes values are treated as the sensitive records.
- There is a good partitioning of the data, and
- There is a large amount of data so that many similar tuples fall into each partition.

Ashwin Machanavajjhala et al. claim that k-anonymity technique does not protect against attacks based on the background knowledge. They also claim that their proposed L-diversity framework gives stronger privacy guarantees. They had also demonstrated that L-diversity and k-anonymity have enough similarity in their structure that k-anonymity algorithms can be modified to work with L-diversity.

2.5. Privacy Preserving DBSCAN Algorithm for Clustering

To discover clusters with arbitrary shape, density-based clustering methods have been developed. These methods typically regard clusters as dense regions of objects in the data space that are separated by regions of low density (representing noise). More formally it can be stated as:

> "A density-based cluster is a set of density-connected objects that is maximal with respect to density-reachability. Every object not contained in any cluster is considered to be noise."

DBSCAN (Density Based Spatial Clustering of Applications with Noise) is one of them, with ability to detect arbitrary shaped clusters. DBSCAN algorithm grows regions with sufficiently high density into clusters and discovers clusters of arbitrary shape in spatial databases with noise. Density is defined as a minimum number of points within a certain distance (ε) of each other. It takes input parameters ε (the minimum radius) and *MinPts* (minimum number of points in any cluster at a distance ε).

DBSCAN searches for clusters by checking the e-neighborhood of each point in the database. If the e-neighborhood of a point p contains more than *MinPts*, a new cluster with p as a core object is created. DBSCAN then iteratively collects directly density-reachable objects from these core objects, which may involve the merge of a few density-reachable clusters. The process terminates when no new point can be added to any cluster. All the non-core objects which are not in the neighborhood of any of the core objects are labeled as noise. DBSCAN doesn't need the number of final clusters to be given in advance. There are several clustering algorithms are available. They are applicable to specific type of data, but DBSCAN is applicable for all types of data and the clusters obtained by DBSCAN are similar to natural clusters.

The sharing of data has been proven beneficial in data mining applications. However, privacy regulations and other privacy concerns may prevent data owners from sharing information for data analysis. To resolve this challenging problem, data owners must design a solution that meets privacy requirements and guarantees valid data clustering results.

2.6. Privacy Preserving Clustering Transaction Perturbation

Most of the data randomization methods focus on distorting the existing attributes of the sensitive transaction in order to preserve the privacy. Such distortions reduce the analytical value of the database or transactions. This problem is especially challenging because of the discrete nature of the attributes corresponding to presence or absence of items. In order to deal with this issue, the

randomization technique needs to be modified slightly. Instead of adding quantitative noise, random items are dropped or included with a certain probability. The perturbed transactions are then used to compute the *Similarity of Dissimilarity* for the clustering purpose.

Although this method can become a simple alternative to the multi-party preserving data mining approach using a mutually agreed upon data encryption protocol; In some functionalities of the data mining process but is not suitable for in case of the data clustering because the transaction perturbation causes over clustering or under clustering problem.

Adding the fake transactions to the database

Some privacy preserving techniques focus on a particular data mining functionality like the Association Rule Mining, Data Classification or the Data Clustering. Adding fake transactions to the database addresses the Privacy Preservation Association Rule Mining.

This technique **[14]** suggests addition of some fake transactions in the database instead of adding quantitative noise to the attributes of the sensitive transactions. Such sanitization of affects significantly the *"Support and Confidence"* of an Association Rule in the database, so the adversary cannot mine the sensitive rules with a reliable *Support and Confidence,* from the perturbed database. This approach is very similar to the *Transaction Perturbation* technique that is discussed earlier. So it shares the advantage and disadvantage with that.

2.7. Privacy Preserving Clustering by EM-Mixture model

Generally distributed data processing and the knowledge discovery is viewed as an optimal solution because the tradition method of building a centralized data warehouse is costly. In the same way the distributed data mining techniques seem better than the centralized data processing as they offer low cost and saving of processing time through use of the parallel computations in a distributed system. The distributed data processing also requires lesser data storage capacity as the user is not required to integrate data from all sources. It also demands lesser human efforts as that is required in the central data repository to integrate data from various sources.

Along with saving of time, cost and the human efforts, the data privacy and security concerns provide another motive for distributed knowledge discovery. Usually the data exits in the distributed form because it has been collected or produced by different parties. Although it is possible to control the

data leakages through the mutual contracts regulations, yet it the privacy concerns can reduce or block the release of the data. In presence of such data secrecy concerns, sometimes it seems more secure and beneficial to preserve data privacy instead of getting benefits of the global data computing. In such situation, the distributed data processing (computing) seems more viable than building a centralized data repository because no single party can be trusted by all of the data owners.

For example a corporation may cluster its customers to identify different groups to target in marketing campaigns. Now imagine that a multinational corporation would like to develop a global advertising, but privacy laws may prevent transferring of the customers' data across borders. Clustering within each country, doesn't give the knowledge needed to develop a global campaign. Distributed clustering is the only solution, provided that it can be done without violating the privacy laws those restrict flow of customer's data across the borders.

Xiaodong Lin, Chris Clifton and Michael Zhu claim that they have presented a clustering method based on expectation maximization that limits the disclosure of data between sites in a distributed environment. Specifically, the values of individual data items are not disclosed. No information can be traced to a specific site.

They used the expectation maximization (EM) mixture model to perform privacy preserving clustering on the distributed data **[15]**. The expectation maximization (EM) algorithm is an iterative method based mainly on the maximum likelihood principle. Since Dempster, Laird, and Rubin's celebrated paper on the EM algorithm (Dempster et al. 1977), it has become a very popular method in the Artificial Intelligence and Statistics community.

This method controls data sharing, preventing disclosure of individual data items or any results that can be traced to an individual site. EM mixture clustering iterates over the data, producing a new set of cluster Centroids, at each time. Over time, these converge to good cluster centers. The authors show that this can be done without revealing individual data points and without revealing which portion of the model came from which site.

The basic idea is that each iteration can be broken into a sum of values corresponding to the partitions of the data. Each partition can be computed locally, based on the local data points and global information from the previous iteration. The global sum is then computed without revealing the individual values. This provides sufficient information to compute the global information needed for the next iteration. Once this process converges, the individual sites can use the resulting model to determine in which cluster their data values lie.

2.8. Privacy preserving k-means Clustering based on additive secret sharing

Mahir Can Doganay et al, presented a privacy preserving k-means Clustering based on additive secret sharing **[16]**. They consider a distributed scenario in which the data is vertically partitioned (different attributes for the same entity can be stored at different sites). In this case each site has a different projection of the database.

The authors choose the popular k-means clustering algorithm and proposed a new protocol for distributed privacy preserving k-means clustering. Instead of using computationally costly public key encryption schemes, they utilize additive secret sharing as a cryptographic primitive to implement a secure multiparty computation protocol in order to do privacy preserving clustering.

The Security of the proposed protocol relies on secret sharing of the necessary data in clustering the distributed data without revealing the private values of individual records at each site. It is assumed that there exist authentic and confidential communication channels. Such channels can be implemented with a combination of symmetric and public key cryptography.

The authors use Euclidean distance to compute the K-Means from the distributed data. Since the data is vertically partitioned, each party can compute part of the distances between each of the *n* entities in the dataset and the cluster means. In Euclidean distance the square of the total distance between an entity and a cluster mean is the sum of the squares of the sub-distances computed at the subspaces of each party:

$$\|x_i - \mu_c\|^2 = \sum_{p=1}^{r} \|x_{ip} - \mu_{cp}\|^2.$$

However, the parties cannot reveal their sub-distances in order to compute the sum of them, since the local sub-distances may contain private information. Therefore they are computed through the "Secure Closest Cluster Computation Algorithm" without revealing the individual sub-distances. The only information that a party will learn after the algorithm is:

- The final mean (μ_{ci}) for each cluster C ϵ {1, . . . , k}.
- The cluster index for each entity J ϵ {1, ..., n}.

The authors argue that the communication and computation cost of their protocol is considerably less than the state of the art which is crucial for data mining applications.

2.9. A Hybrid Data Transformation Approach for Clustering Categorical Data

A.M. Natarajan, R.Rajalaxmi et al. **[17, 18]** propose a hybrid data transformation approach for privacy preserving clustering of categorical data. In the proposed method, the categorical data (or attribute) is converted into binary data and it is transformed using geometric data transformation method. Then, the transformed data can be Clustered using conventional algorithm like K-means, to ensure privacy.

The Hybrid Data Perturbation Method (HDP) combines the strength of the existing methods like Translation Data Perturbation (TDP), Scaling Data Perturbation Method (SDP), and Rotation Data Perturbation Method (RDP). The authors claim that their proposed method we can preserve the original data and also the clustering accuracy when the categorical data is under processing.

2.10. Privacy-preserving Distributed Clustering using Generative Models

Srujana Merugu and Joydeep Ghosh **[19]** present a framework for clustering distributed data in unsupervised and semi-supervised scenarios, taking into account privacy requirements and communication costs. Rather than sharing parts of the original or perturbed data, they instead propose to transmit the parameters of suitable generative models built at each local data site to a central location. A fundamental assumption of their framework is that there is an (unknown) underlying distribution that represents the different datasets and it is possible to learn this unknown distribution by combining *high-level information* from the different sources instead of sharing individual records.

In this framework, the parties owning the individual data sources independently train the generative models on the local data and send the model parameters to a central combiner that integrates the models. This limits the amount of interactions between the data sources and the combiner and enables the data miners to formulate the distributed clustering problem in a general as well as tractable form.

They mathematically show that the best representative of all the data is a certain "mean" model, and then empirically show that this model can be approximated quite well by generating artificial samples from the underlying distributions using Markov Chain Monte Carlo techniques, and then fitting a combined global model with a chosen parametric form to these samples.

The authors claim that the results of their proposed framework show that high quality distributed clustering (Horizontal Distribution) can be achieved with little privacy loss and low communication cost.

Moreover this algorithm is applicable to a wide variety of data types and learning algorithms, so long as they can provide a generative model. **[20]**

2.11. Top / Bottom-Coding

This method is similar to the Generalization technique. It sets an upper limit (top-code) and a lower limit (bottom-code) on quantitative variables. Any attribute having value greater than the upper limit is replaced by the *upper limit* or is not published on the micro-data file at all **[21]**. Similarly, a bottom-code is a lower limit on all published values for a variable. Different limits may be used for different quantitative variables, or for different subpopulations.

For example, the record of an individual with an extremely high income would not contain his exact income but rather a code showing that the income was over $100,000. Similarly the low-income records would contain a code signifying the income was less than $0. In this example $0 is a bottom-code and $100,000 a top-code for the sensitive or high visibility field of income. The advantages and disadvantages of this approach are similar to that of the Generalization technique in the K-Anonymity method.

After having in mind the definition of the Data Clustering along with the precautionary measures for privacy preserving clustering, we have discussed some of the existing privacy preserving Data Clustering techniques. Now we are ready to discuss the suitability of the techniques.

The above study of the various privacy preserving techniques enable us to conclude that in general, there are two approaches for designing privacy-preserving data clustering algorithms. In the first approach is the data miner is allowed to apply the algorithms on the complete database but before such permission some sort of transformations is used to perturb the dataset in order to preserve its privacy. This approach for designing privacy-preserving clustering algorithms is taken by several researchers as mentioned earlier [3, **4**, 5 and others].

The second approach to design privacy preserving Clustering algorithms is to reuse the existing algorithms that were actually developed for the secure-multiparty computation. These algorithms accept this limitation or restriction that it is not feasible or not allowed to allow access to the complete database. Especially in some cases it becomes nearly impossible to gather the distributed data into a single repository or very large database (VLDB) like a Data Warehouse. Hence the data miners have to adopt an alternative framework to perform the distributed computation securely and accurately without accessing directly the data partitions at various sites. This frame work has been proved not only secure and efficient

but as well as economical. *Data Encryption* (Cryptographic techniques) and Data transformation techniques fall in this category. The work of [8, 9, 10 and 11] is based on the latter approach. Although these techniques require complex and time taking processing on the data for the encryption, they are suitable choice PPC especially when:

- The Clustering process is being performed in a distributed environment—multiple parties are feeding the database or data warehouse.
- An organization or enterprise wants to out-source the Data Clustering tasks.

We can deduce the following facts from the above discussion:

a) Like the Association Rule Mining, accurate and reliable Data Clustering also requires the *Complete Access to the database* and *Preservation of the Attributes of the Transaction.*

b) Data perturbation or sanitization affects the attributes of a transaction to preserver its privacy and resultantly reduces the analytical value of the data.

c) The transaction perturbation change the original "Similarity or Dissimilarity" among the data object or transactions; therefore the resultant Clustering cannot be considered reliable with the required confidence.

d) Techniques that apply attribute 'Suppression' or 'Generalization' to preserve the privacy of data; significantly affect the results of the applied similarity measures, which play the decisive role in the clustering of the data objects. So analytical value of the *Clusters* will reduce significantly.

e) Attribute 'Suppression' or 'Generalization' process perturbs or hides the *Outliers* in the data that may be of high value in a particular scenario like fraud detection, Credit Card theft detection and in Forensics.

f) The privacy preserving techniques that *SWAP* attributes of the transactions or records not only affects the original *'Similarity or Dissimilarity'* of the existing data objects but the swapping may introduce the "Ghost" or synthetic members in the resultant Clusters as well.

g) The information (in our case the Means of the resultant Clusters) extracted from the "partially disclosed data samples" doesn't represent the complete database. An object that falls in a

Cluster based on a Data partition or sample may fall into a completely different Cluster that is based on the complete database. Hence such Clustering is unreliable for the research purpose.

h) In the Data warehouse environment, normally the database is refreshed periodically by adding/appending the fresh data. In such case the previously generated Clusters becomes invalid. In such scenario newly added data is included in the mining process after the sanitization. This makes the sanitization a regular overhead.

In short we can say that privacy preserving technique that sanitized the transactions or database ultimately disturb the accuracy of the Data Clustering. Nevertheless when the target domain is highly sensitive and require not less than the data perturbation, a balance must be maintained between privacy and the accuracy.

Table 2.1: A Comparison of the Data Privacy Preserving Techniques

Technique Name	Computational Value	Computational Complexity	Suitability for Privacy Preserving Clustering
Privacy preserving clustering by data randomization (perturbation)	Proportional Loss of the computational value	Depends on the randomization Level	Suitable when normal & quick output is required
Privacy preserving clustering by K-anonymity technique		Proportional to the size of the K, L respectively	Used in publically available dataset
Privacy preserving clustering by L-diversity technique			
Privacy preserving clustering by transaction perturbation		Proportional to the size of the target dataset	
Privacy preserving clustering by EM-Mixture model	Depends on accuracy of the EM-Model		Suitable for private distributed dataset
A hybrid data transformation approach for clustering categorical data	Controlled Loss		Suitable in bioinformatics etc.
Privacy-preserving distributed clustering using generative model	Proportional to the complexity of the model		
Linear Data Transformation	Controlled Loss	Doubles the	Suitable when nearly natural clustering output is required

Chapter 3

The Proposed Solution

3.1. Significance of the Privacy Preserving Density Based Clustering

Data Clustering is a method of unsupervised learning in the Data Mining domain. It is a commonly used technique for statistical data analysis in various fields, pattern recognition, image analysis and bioinformatics, forensics and the machine learning etc.

The Data Mining analyzes data from different perspectives and extracts hidden patterns. These analyzed facts and patterns are summarized into useful information. Data Mining is among the emerging fields in the computing technology and connects the three major technical areas-- Databases, Artificial Intelligence and Statistics.

Data Mining is an interdisciplinary field with respect to its application; it attracts the specialists, researchers and interested parties from different areas such as, Artificial Intelligence, Biology, Commerce, Mathematics, Computer Science, Information Systems, Agriculture, Weather Forecast and the Statistics. The Data Mining fulfills the common need of extraction of useful but previously unknown knowledge that can support in decision-making processes.

Advancements in information processing technology and the storage capacity have established the Data Mining as a widely accepted technique in the decision making process. Many organizations, dealing with large databases, are highly dependent on Data Mining in their functioning.

3.1.1. The Need to preserve data privacy in Data Mining

The rapid advancement in information processing technology and availability of affordable large storage capacity, are paving the way for various organizations to adopt the Data Mining technique in their decision making process. More and more organizations are becoming dependent on Data Mining in their day to day business activities. Usage of the Data Mining is resulting better decision-making, providing better service, and achieving greater revenues.

3.1.2. Nature of the Problem

In the past two decades various privacy preserving techniques has been proposed to preserve privacy of the private data; but none of them is considered as an ultimate solution to the privacy issue. Each of the techniques fits to a certain scenario. A privacy preserving technique for a one domain would become inapplicable at all in another scenario. Hence selection of an optimal PPDM technique is a serious issue for the data miners. Data mining is divided into various functionalities, such as Association Rule Mining, Data Clustering, Data Classification and Prediction etc. Normally the Data miners focus on a

particular Data Mining functionality and tailor an existing Data Mining technique to make it privacy sensitive according to the target domain.

During the Pre-Literature review, I consulted work of seminal experts of data mining and found it specific and domain oriented. Here are few citations of the articles related to the subject. Stanley R.M. Oliveira and Osmar R. Zaiane, *'Preserving Clustering by Data Transformation'*. Jun-Lin, *'Privacy Preserving Clustering by K-Anonymity technique'* **[21]**. Stanley R. M. Oliveira et al, *'Privacy Preserving Frequent Itemset Mining'*. **[22]**

3.2. The Proposed Solution

In minimum words, we can say that the DBSCAN requires value added implementation that would preserve privacy of the target dataset up to the desired level. Detail of the implementation is following.

3.2.1. Research Domain-- Data Clustering

Clustering is a process of grouping or partitioning a set of physical or abstract objects or items into classes of similar objects. The objects within a Cluster significantly resemble each other (Intra-Cluster Similarity); while the objects of a cluster are significantly differ from the objects in other clusters (Inter-Clusters Dissimilarity).

Usually the partitioning of object is done by measuring Dissimilarity (or Similarity) using a distance measure like Euclidean, Manhattan, Hamming and Minkowski distances. Clustering is also called *Data Segmentation* or an *Unsupervised Learning* as it does not rely on predefined classes and class-labeled training examples. For this reason, clustering is a form of learning by observation, rather than learning by examples. **[23]**

Clustering is a dynamic field of research in data mining. Various clustering algorithms have been developed and implemented. These can be categorized into Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Model-Based Methods, Methods For High-Dimensional Data (including frequent pattern–based methods), and Constraint Based Methods.

Unsupervised learning deals with designing classifiers from a set of unlabeled samples. A common approach for unsupervised learning is to first cluster or group unlabeled samples into sets of samples that are "similar" to each other. Once the clusters have been constructed, we can design classifiers for each cluster using standard techniques such as Decision-tree learning **[24]**.

The quality of a Clustering technique can be assessed based on a measure of dissimilarity of objects, which can be computed for various types of data, including interval-scaled, binary, categorical, ordinal, and ratio-scaled variables, or combinations of these variable types.

Clustering has many applications, some of them are listed below: such as Customer's behavior analysis, Targeted marketing, Forensics, and Bioinformatics, market research, pattern recognition, data analysis, and image processing.

- In business, Data Clustering can help the businessmen to discover distinct groups in their customer bases and characterize their purchasing patterns.
- In Biology, it can be used to derive taxonomies of the plants and animals.
- In Bioinformatics it is used to categorize the genes with similar functionality, and gain insight into structures inherent in populations.
- Clustering may also help in the identification of areas of similar land use in an earth observation database.
- In Urban planning and development it can help to identify the groups (or Clusters) of houses in a city according to house type, value, and geographic location.
- In Insurance industry, Clustering is used to identify the groups of insurance policy holders with a high average claim cost.
- Clustering can also be used for outlier detection; the outliers are of great interest in some cases like in fraud detection, Credit Card theft detection and criminals capturing (Forensics).
- In image processing, Clustering is a very suitable technique to identify or classify the similar (or different) features of the image under study.
- It can also be used to help classify documents on the Web for information discovery.

Data Clustering Example

They following figure shows a 2-Dimentional plot of customer's data of a Cellular Phone service provider Company, with respect to the customer's locations in Pakistan, showing three Data Clusters. Each cluster "center" is marked with a "+" sign (representing the city).

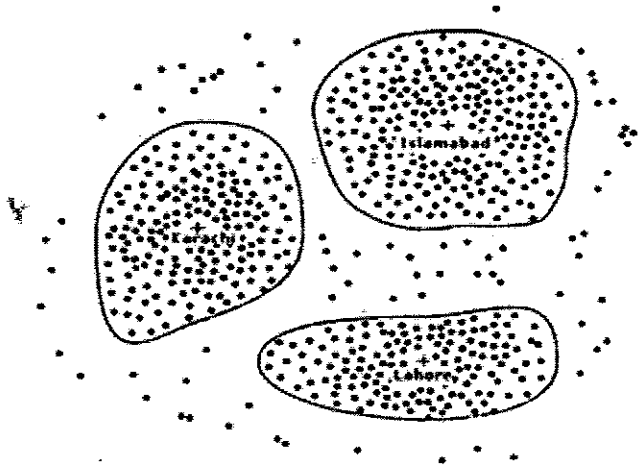


Figure 3.1: Data clustering of customer's data of a company with respect to the customer's location in three majors cities of Pakistan.

3.2.2. Processing nature of the Data Clustering

We can deduce following facts about the Data Clustering from its definition:

- *Complete access to the database:* The Clustering requires complete database scan to determine the dissimilarity or (similarity) among the data items or transactions, so any privacy preserving technique should not restrict the database access for the Data Clustering. In case of restricted or partial access to the target database, the output clusters and their calculated means will not reflect the complete database. Hence they can not be generalized for whole database with the desired accuracy and reliability.
- *Preserving the attributes of the transactions:* In data clustering the attributes or characteristics of an object or transaction are the sole parameters that are used by dissimilarity measures to calculate the dissimilarity (or similarity); so they cause the difference among the objects or transactions; this difference is the base to include (or exclude) an object or transaction in a

particular cluster. Therefore any privacy preserving technique should not disturb the original similarity or dissimilarity among the object or transactions in order to preserve the privacy.

- Similarly the perturbation of the attributes that would distort or eliminate the inherent similarity or dissimilarity among the data objects or transactions is also harmful for the accurate Data Clustering. Such distortion ultimately reduces the analytical value of the data. . Nevertheless when the target domain is highly sensitive and require not less than the data perturbation to ensure the privacy, then a balance must be maintained between privacy and the accuracy.

3.3. Justification of the research work

In the last few decades the Data Mining has been applied in a wide variety of areas, including Financial Data Mining, Text Mining, Web Mining, Healthcare (Bio Informatics), Scientific Data Mining, Data Mining in Oil and Gas industry etc. Along with this there has been much interest recently on using data mining for Fraud-Detection and Counter-Terrorism applications.

In the last two decades, knowledge discovery and data mining tools have been used mainly in experimental and research environments, but now we observe abundance of the sophisticated tools, which are in use of the mainstream business users. New tools hit the market nearly every month. "The Meta Group estimates that the market size for data mining market will grow from $50 million in 1996 to $800 million by 2000" **[25].**

An overview of the current trends in information processing technology reveals the fact that scientist, researchers and Leaders of the computing industry are focusing on the issue of data privacy. Considerable research work is underway in this direction. But Data Mining being too area specific still offers a lot of opportunities to the young scholars to invent such data mining techniques that are sensitive to the issue of data privacy.

3.4. Employed Data Clustering Method

Clustering is a dynamic field of research in data mining. Various clustering algorithms have been developed and implemented. These methods and techniques are categorized as below:

- **Partitioning Methods**
  - K-Means/Median/Mode Clustering, K-Medoids Clustering
  - CLARA (Clustering LARge Applications)
  - CLARANS (Clustering Large Applications based upon RANdomized Search)
- **Hierarchical Methods**
  - Agglomerative Clustering {Chameleon, AGNES (AGglomerative NESting) }
  - Divisive Clustering { BIRCH(Balanced Iterative Reducing and Clustering Using Hierarchies), DIANA (Divisive ANAlysis) }
  - ROCK (RObust Clustering using linKs)
- **Density-Based Clustering Methods**
  - DBSCAN (Density-Based Spatial Clustering of Applications with Noise)
  - OPTICS (Ordering Points to Identify the Clustering Structure)
  - DENCLUE(DENsity-based CLUstEring) Clustering Based on Density Distribution Functions
- **Grid-Based Methods**
  - STING (STatistical INformation Grid)
  - Wave-Clustering (Clustering Using Wavelet Transformation)
- **Model-Based Methods**
  - Expectation-Maximization (EM) based Clustering
  - COBWEB and CLASSIT-- an extension of COBWEB
  - Self-organizing feature maps (SOMs)
- **Methods for High-Dimensional Data (including frequent pattern–based methods)**
  - CLIQUE (CLustering InQUEst)

- PROCLUS (PROjected CLUStering)

The selection of clustering algorithm depends both on the type of data available and on the particular purpose of the application. I selected the Density Based Clustering approach for my research work and I shall tailor the DBSCAN algorithm according to requirements of the application. The Density based clustering approach has following benefits over the typically used clustering techniques such as K-Mean, and CLARANS etc.

a. Most partitioning methods cluster objects on the basis of the distance between objects. So these methods can find only spherical-shaped clusters and encounter difficulty at discovering clusters of arbitrary shapes. The Density Based Clustering techniques can discover clusters of arbitrary shapes along with the spherical shape clusters.

b. Typically used clustering techniques such as K-Means, K-Medians etc. require the user to input the number of clusters to be formed. The necessity for users to specify the number of clusters, in advance is a disadvantage. As if the user specifies small number of clusters, the clustering algorithm may merge two or more sufficiently different clusters (Coarse Clustering); while if the user specifies large number cluster then the algorithm may split a single cluster into multiple smaller clusters, just to accommodate the user specified number of clusters.

c. Data Clustering techniques that use the Central Tendency Measure (Mean, Median etc) to cluster the objects are sensitive to the extreme values or the out lairs, as these extreme values significantly disturb the central tendency measure and distort the clustering result. The Density based clustering approach does not rely on such a measure so they successfully cluster the data with extreme values.

3.5. Employed Data Privacy Preserving Technique

A simple but effective encryption technique is the ***Linear Transformation*** or attribute mapping; in it an attribute is mapped to a pre-calculated artificial value. Such transformation hides the original attributes and prevents an adversary to decrypt the compositions of transactions easily.

Advantages:

a) *Linear Transformation (encryption)* hides the original records completely, so it eradicates the inference problem. If the encryption method is secure enough then the original individual records cannot be recovered from the encrypted database.

b) If the encryption is accurately reversible then it the results obtained from the encrypted database can be decrypted to get the original results without losing any information. So the analytical value of the records can be preserved.

c) The proposed technique can be used on a limited set of variables, without disturbing responses for the non-sensitive fields.

d) Non-trusting parties can jointly compute functions of their different inputs while ensuring that no party learns anything but the defined output of the function. They can do so using a suitable encryption protocol.

e) Data Mining is complex tasks that require heavy resources and expertise. The companies that lack such resources can Out Source their data mining tasks. In such case encryption techniques are highly preferable.

f) The user encrypts the data before allowing processing on it; this gives him more satisfaction in terms of database privacy.

g) The procedure is not limited to continuous variables; categorical variables (such as race, Gender, Occupation) can also be swapped.

3.6. Delimitations and Assumptions of the research work

a) Data Mining requires a single, separate, clean and consistent source of data. A Data Warehouse provides data source having most of the required features. So to perform the data mining activities, it is assumed that an efficient Data Warehouse or at least a clean and consistent dataset is available.

b) The Privacy Preserving Data Mining through the Linear Transformation can be applied on various functionalities of Data Mining like the Data Classification, Association Rule Mining and Outlier's Analysis etc. But this research work is limited to the Data Clustering only; as it is one of the most famous and important functionalities of Data Mining. After the successful implementation for Data Clustering, we can extend this approach to the other data mining functionalities according to their requirements.

c) Since the aim this research work is to preserve the data privacy using any suitable and established Data mining technique so it is assumed that the data mining technique that is being used; is giving the required results accurately.

3.7. The Research Method

To conduct a research work a researcher can adopt various research methods like the Experiment, Survey, Simulations, Case-Study, Benchmarking or the Ethnography depending on the nature of the research and suitability of the research method. In order to demonstrate the preservation of privacy in the Data Clustering through data transformation, it is required to conduct an ***'Experiment'*** on the synthetic or the original data. I shall develop an software application using C# to provide the Graphical User Interface (GUI) that shall enable the end user to Cluster his database through the simple interactive commands. The GUI shall be connected with ORACLE the famous DBMS available in the market. During the implementation phase I shall use the syntactic data that shall be generated especially for the application.

The synthetic data shall represent the database of a Bank that is in need to cluster its customers according to their "Loyalty" with the Bank. The Loyalty of a customer is based on his profile like his

Income Group, Period of subscription (In years), Age Group, Profession etc. the bank is planning to launch some new products including Car-Financing Scheme, Personal Loan, and House-Building Loan etc. The Bank management has an intension to introduce these new products to wisely selected groups of the customers instead of sending the advertisement material to every customer. Since the clustering process requires intensive and unbiased analysis of Customer's data that is very difficult to do manually. So the bank decides to hire services of the Data Miners. Here comes the need to preserve privacy of data; as the bank is morally and legally bound to maintain privacy and secrecy of record of every customer, hence the bank is in need of *Privacy Preserving Grouping/Clustering* of its customers. The proposed application will fulfill the bank demands by employing a simple but reliable Privacy Preserving Data Clustering technique to group the customers.

As for as other research methods are concerned; they are not suitable for this specific problem. Like the *'Survey'*, is not applicable because according to my knowledge this approach is not adopted or recommended in past, by any researcher or data mining expert.

'Simulation' is not a possible choice because in Data Mining the algorithms are applied on the synthetic data, if the original data is not available and the experiment is done on small scale (with easy to handle amount of data) if it is not feasible to work on complete data.

'Case-Study' is not applicable in our case, as it is required to prove efficiency of the newly propped privacy preserving clustering technique through its implementation in the specified domain, rather than observing it in another domain.

We cannot use the *'Bench Marking'* approach because the Data mining is too domain specific and requires considerable customization of algorithm and the privacy preserving technique according to the level of required privacy. Similarly the Ethnography is not applicable in this case.

Chapter 4

Implementation

4.1. Selection of the clustering domain

To show the applicability and efficiency of a clustering algorithm it is required to implement it on a dataset of a potential data clustering domain. National Databases, Retail marketing record, Customer's credentials of a Banking, Insurance data, medical history of the patients and weather forecasting data etc. have been among the potential target domains for the data clustering or data mining.

In this research work the proposed value added privacy preserving density based clustering algorithm is applied on dataset related to the Banking sector. Selection of the banking sector dataset was result of the following two motivating factors.

- First, the banks are among the leading sectors, those are rapidly moving towards the Data Digitization, Electronic Data Processing (EDP), Database Management Systems (DBMS), Data Warehousing (DWH), Data Mining (DM) and even towards the Business Intelligence (BI) etc. therefore it seems prudent to address the banking sector first.
- Second, the financial institutions have to bee over-sensitive towards privacy of the public data; therefore there is more attraction in a privacy preserving clustering algorithm for these sectors.

As it is mentioned above that the financial institutions are sensitive towards privacy of the individual's record, so they hesitate to share the dataset with the research students. To overcome this obstacle it was decided to use the synthetic data according to metadata of dataset of a bank. The synthetic data shall represent the database of a Bank that is in need to cluster its customers according to their "Loyalty" or profitability to the Bank. The Loyalty of a customer is based on his credentials like his Income Group, Period of subscription (In years), Age Group, Profession etc.

The bank is planning to launch some new products including Car-Financing Scheme, Personal Loan, and House-Building Loan etc. The Bank management has an intension to introduce these new products only to the wisely selected groups of the customers instead of sending the advertisement material to its every customer. Since the clustering process requires intensive and unbiased analysis of Customer's data that is very difficult to do manually. So the bank decides to hire services of the Data Miners. Here comes the need to preserve privacy of data; as the bank is morally and legally bound to maintain privacy and secrecy of record of every customer, hence the bank is in need of *Privacy Preserving Grouping/Clustering* of its customers. The proposed application will fulfill the bank demands by employing some simple but reliable Privacy Preserving Data Clustering techniques to group the customers without breaching privacy of the individual record.

4.2. Acquisition of the Dataset

The synthetic dataset was generated through the multiple means like the online data generating software on the Internet, the Procedural Language / Structural Query Language (PL/SQL), spread sheets etc. The websites **[26, 27]** were used to generate the customer's credentials like Customer's Age (in years), Annual Salary (in thousand), Educational Qualification (in years) etc. these numeric values were transformed to the suitable string literal (labels) by using the spread sheets or PL/SQL queries. Some of the credentials like the Customer's IDs were generated through the spread sheets of Microsoft EXCEL. Initially a dataset of 10,000 customers was generated but later only 5,000 records were used in the clustering process. This data reduction was aimed to reduce the computational load and to handle safely the well known problem of the "Curse of Dimensionality" in the clustering process.

4.3. Data Pre-processing

The Data Mining process usually deals with databases having huge size (often several gigabytes or more). These real-world databases are highly vulnerable to noisy, missing, and inconsistent data due to the large size and their likely origin from multiple, heterogeneous sources. The inconsistencies, noise and incompletion of data etc. result into the low-quality data which eventually lead to the low-quality data mining results. In order to overcome the data inconsistency problem and to improve the data quality the data need to be preprocessed. This data preprocessing not only improves the mining results and makes the mining process easier but as well as increases efficiency of the data mining algorithm. There are a number of data preprocessing techniques; some of them are briefly mentioned in the following:

- **Data Cleaning**

Careful inspection of the database or data warehouse to identify and select the attributes or dimensions that are to be analyzed by data mining techniques, shows that multiple records or tuples are:

- ✓ *Incomplete* (lacking attribute values or certain attributes of interest, or containing only aggregate data)
- ✓ *Noisy* (containing errors, or outlier values that deviate from the expected), and
- ✓ *Inconsistent* (e.g. containing discrepancies in the department codes used to categorize items).

Incomplete, noisy, and inconsistent data are commonplace properties of large real world databases and data warehouses. Incomplete data can occur for a number of reasons. Attributes of interest may not always be available, such as customer information for sales transaction data. Other data may not be included simply because it was not considered important at the time of entry. Relevant data may not be recorded due to a misunderstanding, or because of equipment malfunctions. Data that were

inconsistent with other recorded data may have been deleted. Furthermore, the recording of the history or modifications to the data may have been overlooked. Missing data, particularly for tuples with missing values for some attributes, may need to be inferred.

Data cleaning routines work to "clean" the data by filling in missing values, smoothing noisy data, identifying or removing outliers, and resolving inconsistencies. If users believe the data are dirty, they are unlikely to trust the results of any data mining that has been applied to it. Furthermore, dirty data can cause confusion for the mining procedure, resulting in unreliable output.

- **Data Integration**

Data mining often requires data integration—the merging of data from multiple sources into a coherent data store, such as a data warehouse. These sources may include multiple databases, data cubes, or flat files. The data may also need to be transformed into forms appropriate for mining. The data integration involves number of issues such as:

- ✓ *Entity identification problem* (multiple data sources may contain different attributes for the same entity)
- ✓ *Attribute Redundancy* (An attribute such as annual income, may be redundant if it can be "derived" from another attribute or set of attributes). Inconsistencies in attribute or dimension naming can also cause redundancies in the resulting data set.

Various techniques are used to integrate the data such as use of the metadata while integrating the various schemas and removal of the attribute redundancy by the Correlation analysis.

- **Data Transformations**

In data transformation, the data are transformed into appropriate forms for mining. Data transformation can involve the following:

- ✓ *Smoothing: it* removes the noise from the data by normalizing the extreme or outlying values. Binning, Regression, and Clustering are among the data smoothing techniques.
- ✓ *Aggregation technique* applies the summary or aggregation operations to the data to determine the aggregate or derived attribute. For example, the monthly income data may be aggregated so as to compute the annual income of a customer. This step is typically used in constructing a data cube for analysis of the data at multiple granularities.
- ✓ *Generalization of the data*, where low-level data are replaced by higher-level concepts through the use of concept hierarchies. For example, the values for numerical attributes, like age, may be

mapped to higher-level concepts, like youth, middle-aged, and senior. The age can also be mapped to the age-intervals like 18 to 25 years, 26 to 35 years etc.

- ✓ Normalization, where the attribute data are scaled so as to fall within a small specified range, such as -- 18 to 25 or 26 to 35 etc.
- ✓ *Attribute* construction (or feature construction), where new attributes are constructed and added from the given set of attributes to help the mining process.

- **Data Reduction**

The Data Reduction techniques allow us to mine a reduced data yet producing the same (or almost the same) analytical results. These techniques can be applied to obtain a reduced representation of the data set that is much smaller in volume, yet closely maintains the integrity of the original data. This technique can reduce the data size by aggregating, eliminating redundant features.

- ✓ *Data Cube Aggregation: it* applies the aggregation operations to the data to construct a data cube.
- ✓ *Attribute Subset Selection*: this method allow us to trim or remove the irrelevant, weakly relevant or redundant attributes or dimensions.
- ✓ *Dimensionality Reduction*: this technique is used to encode the string / character data types to the numeric or binary format. For example the gender (male or female) of customer can be encoded as 0 and 1 respectively. Such transformation not only significantly reduces the dataset size but also makes it easier to process the data.
- ✓ *Numerosity Reduction*, where the data are replaced or estimated by alternative, smaller data representations such as parametric models (which need store only the model parameters instead of the actual data) or nonparametric methods such as clustering, sampling, and the use of histograms.
- ✓ *Discretization and Concept Hierarchy Generation*, where raw data values for attributes are replaced by ranges or higher conceptual levels. Data discretization is a form of numerosity reduction that is very useful for the automatic generation of concept hierarchies. Discretization and concept hierarchy generation are powerful tools for data mining, in that they allow the mining of data at multiple levels of abstraction.

After a brief introduction of the data pre-processing methods and techniques it seems logical to mention that in the undergoing research work there was a little need to pre-process the data because of the following reasons.

- ✓ The dataset was generated synthetically by using different data generation utilities or software; so there was no chance for the data noise, inconsistencies etc. such as the missing values, inconsistent attribute etc.
- ✓ The dataset was generated by similar sources; therefore there was no issue of the data integration etc.

it seems worthy to mention that the Data Reduction techniques were applied on the synthetic data in following ways:

- ✓ The financial institutions like the banks maintain dozen of attributes or credentials of their customers to fulfill requirements of the everyday transactions; but all of the attributes are not of significance for calculation of profitability or "loyalty" of a customer. Therefore attributes like Contact numbers, residential address etc. were removed from the dataset to be analyzed. Only seven attributes (Annual Income, Subscription Period, Profession, Age, Education, Account Type and Marital Status) of a customer's record were included in the reduced dataset. This *Attribute Subset Selection* not only reduced size of the dataset but also made it easier to process the data. The following figure shows a specimen of a Bank Account Opening Form that the banks use to collect the customer's data.
- ✓ The *Dimensionality Reduction* technique was also used to encode the string values to the numeric or binary format. For example the three Account Types (Current, Saving or Joint Saving Account) were encoded to the three numeric values (0, 1 and 2) respectively. Similar transformation was applied on Marital Status. These transformations not only significantly reduced the dataset size but also made it easier to process the data.
- ✓ *Data Normalization technique* was also used while generating the attribute values. For example it was made necessary for every bank customer to not have an age below the eighteen years (the legally defined minimum age of an adult in Pakistan). So the first age interval / range, was defined 18 to 25 years. The second range was defined as 26 to 35 years and so on.

F-559
(Revised-Jan' 07)

National Bank of Pakistan

APPLICATION FOR ACCOUNT OPENING/ ISSUANCE OF TERM DEPOSIT RECIEPT

Date
Account No.
TDR No.

Branch: **Branch Code:**

Type of Account Please tick relevant box						
		Current		BBA		Call Deposit
		PLS SNTDR		PLS Savings		NIDA
		PLS Term		Premium Saver		Premium Amdani
		FC Current		FC Saving		FC Term

Nature of Account Please tick relevant box				
		Individual (Single/Joint)		**Sole Proprietorship**
		Limited Company (Public/Private)		**Partnership (Registered/ Unregistered)**
		Govt. Institution (Federal/Provincial)		**Corporate Body (Incorporated Unincorporated)**
		Trust		**Association/ Club/Society**

Currency		**Pak Rs.**		**US $**		**UK£**		**EURO**		**JY**
Initial Deposit / Total Amount of TDR	**Rs.**_____**(Rupees**_____)									

Please open an account/issue deposit receipt as requested above.* I/We have read and understood the rules governing the conduct of account.* The required documents are enclosed. (* Delete which ever is not applicable).

Particulars of Account (For Personal A/c.)				
Photograph for illiterate person & A copy of CNIC	**Title of Account**			
	Address			
	CNIC No.			
	NTN (if available)			
	Profession:			
	Telephone	Off	Fax	Res.
		Mobile	Email:	

Next of Kin (for Individual / Single account)				
Name & address of person/next of kin to be contacted for ascertaining my/our whereabouts after the expiry of three years from the date of last operation in my account/last communication from me/us to the Branch, to inform him/her of the existence of my/our account when I/We were not available at the given address.				
Name				
Father Name				
CNIC No.				
Address				
Telephone	Off	Fax	Res.	
	Mobile	Email:		
Attached Copy of CNIC of the person nominated as Next of Kin YES / NO				

Operating Instructions	The account will be operated by (please tick relevant box)		
	Singly	**Jointly**	**Either/ Survivor**

Special Instructions	**Dispatch of Statement of Account** **Dispatch: Monthly/Quarterly/Half Yearly at my / our request/Hold the statement.**			
Zakat Deduction		**Yes**	**No**	**If no please submit affidavit/declaration as per Zakat Rules)**

Account Holder "A" Signature	Account Holder "B" Signature	Account Holder "C" Signature

Figure 4.1: Specimen of the Bank Account Opening Form

4.4. Preserving Privacy of the Data

The Data Mining can benefit the humans in many ways but there is also a negative side to this technology, since it could be a threat to the privacy of individuals and civil liberties. Therefore the future Database Systems must include the responsibility for the privacy of data they manage as a founding principle. Similarly the other data processing applications like Online Analytical Processors (OLAP) or the Data Warehousing applications must be sensitive to the issue of the Data Privacy. A fruitful direction for future data mining research will be the development of techniques that incorporate privacy concerns; in other words we are in need of the Privacy Preserving Data Mining (PPDM).

As it has been mentioned earlier that the bank is morally and legally bound to maintain privacy and secrecy of record its all customers, hence the bank is in need of *Privacy Preserving Grouping/Clustering* of its customers. To fulfill demands of data privacy by the bank some simple but reliable Privacy Preserving Data Clustering techniques are employed during the clustering process. The brief description of the techniques is following:

- **Attribute Suppression / Attribute Filtering**

Attribute Filtering is privacy preserving method In which value of the sensitive or restricted attribute of an individual record, is removed or blocked completely. For example, the unique identifiers such as Employee ID, National / Social Security Number, Father's / Husband Name are removed from the records. This technique has been employed on the data under process during the research work. The following list shows some of the customer's credentials ~~those~~ that were not allowed to be used in the clustering process.

- ✓ Name, Father / Mother / Husband Name
- ✓ Contact Numbers
- ✓ E-mail address
- ✓ Postal addresses
- ✓ Exact Date of Birth
- ✓ National Identity /Social Security Card Number
- ✓ National Income Tax Number
- ✓ Name and details of his next of Kin

It is worth to mention that this sanitization still does not ensure the privacy of individuals in the data. A study estimated that 87% of the population of the United States can be uniquely identified using the seemingly innocuous attributes of gender, date of birth, and 5-digit zip codes.

- **Attribute Generalization**

Attribute Generalization is another privacy preserving technique in which the attribute values are generalized to a range in order to reduce granularity of the data representation. For example, the exact date of birth (with day, month and year) could be generalized to a range such as year of birth [such replacement is also known as *Mapping to Higher-Level Concepts*]. Similarly the exact salaries can be generalized to salary range like a value "5 to 10 thousand" can be substituted in place of the salaries like 5000, 7000 or 9000 to reduce the risk of identification. Another way to generalize the salaries is use of the annual salary in place of the monthly salary. This privacy preserving approach also has been employed on the target data in the clustering process.

- ✓ The exact date of birth has been generalized to the one of the predefined age intervals such as 18 to 25 years, 26 to 35 years and so on. The replacement made it difficult for an adversary database user to recognize the record of and individual.
- ✓ The monthly income of an individual customer is kept hidden in the target database by replacing it with the annual income of the customer.
- ✓ The exact educational qualification such as *Maters in Business Studies* or *Masters in Computer Studies* has been generalized to Masters or Post Graduate labels.

- **Linear Data Transformation**

A simple but effective encryption technique is the *Linear Transformation* or attribute mapping; in it an attribute is mapped to a pre-calculated artificial value. Such transformation hides the original attributes and prevents an adversary to decrypt the compositions of transactions easily. The following paragraphs cover few advantages and disadvantages of the data encryption techniques to preserve data privacy.

Advantages:

a) Encryption hides the original records completely, so it eradicates the data inference problem. If the encryption method is secure enough then the original individual records cannot be recovered from the encrypted database.

b) If the encryption is accurately reversible then the results obtained from the encrypted database can be decrypted to get the original results without losing any information. So the analytical value of the records can be preserved along with preservation of data privacy.

c) It can be used on a selected set of one (or more) variables, without disturbing the responses for non-sensitive and non-identifying fields.

d) Non-trusting parties can jointly compute functions of their different inputs while ensuring that no party learns anything but the defined output of the function. They can do so using a suitable encryption protocol.

e) Data Mining is complex tasks that require heavy resources and expertise. The companies that lack such resources can Out Source their data mining tasks. In such case encryption techniques are highly preferable.

f) The user encrypts the data before allowing processing on it; this gives him more satisfaction in terms of database privacy.

g) The procedure is not limited to continuous variables; categorical variables (such as race, Gender, Occupation) can also be swapped.

Disadvantages:

a) The data encryption technique requires complex processing on the data for its implementation so it takes considerable time and requires intensive processing.

b) It requires sufficiently large amount of space (at least double space is required) for its implementation, as the user has to maintain the encrypted data along with the original.

c) It is difficult to implement, as the encryption technique having sufficient strength requires expertise in the data encryption.

This privacy preserving approach also has been employed in the clustering process. The chart of loyalty given on the following page shows the suppressed credentials of a customer along with the linear data transformation codes. According to the chart the "Martial Status" of a customer has been transformed into three distinct codes—1 for single, 2 for Widowed / Divorced and 3 for a customer having status "Married". Cleary the linear transformation hides sensitive attribute -- *marital status* to the persons who are analyzing the data through the mining process.

It is necessary to mention that the transformed values not only preserve the data privacy but they are representing the customer's loyalty or profitability to the bank. Definitely the transformation need be done meticulously to serve the dual purposes. In the on going clustering process since the data under use is synthetic therefore the linear transformation has been kept simple and to minimize the process load of data clustering. This transformation can be changed / strengthen according to the required level of privacy for the target database. The linear transformation codes can complex according to distribution of values for a single attribute of an individual record. Transformation of the other attributes is easy to understand in the following loyalty chart.

Table 4.1: The Linear Transformation Table to Encrypt the Customer's Data

Chart of Customer's Loyalty

Customer's ID (Sequential Number (1 to 5, 000))

It is used as a Primary Key to uniquely identify the customer's Record

Marital Status (1 to 3)

Single	1
Widowed / Divorced	2
Married	3

Income Group (Loyalty 1 to 6)

The Bank assigns income groups to its Customers according to their Annual Income

Income (Annual)	Loyalty
Below or equal to 100, 000	1
Above 100, 000 but Less than equal to 300, 000	2
Above 300, 000 but Less than equal to 500, 000	3
Above 500, 000 but Less than equal to 700, 000	4
Above 700, 000 but Less than equal to 1, 000, 000	5
Above 1, 000, 000	6

Age Group (Loyalty 1 to 5)

The Customer's Age also play a significant role in assessing loyalty of a customer. The Bank follows following criteria.

Age	Loyalty
Between 18 to 25 Years	1
Between 25 to 35 Years	2
Between 35 to 50 Years	3
Between 50 to 60 Years	4
Above than 60 Years	5

Subscription Period (Loyalty 1 to 4)

The Bank assigns high value to the customers with long subscription period.

Subscription Period	Loyalty
Less than 2 years	1
Between 2 to 5 years	2
Between 5 to 8 years	3
Above than 8 years	4

Customer's Education (Loyalty 1 to 4)

Highly educated persons are more trustworthy and profitable for the bank. So it considers them more loyal.

Education	Loyalty
< = 10 years of Schooling	1
> 10 but < = 14 years of Schooling	2
> 14 but < = 18 years of Schooling	3
Above than 18 years of Schooling	4

Customer's Profession (Loyalty 1 to 5)

The Bank assigns high value to the customers with long subscription period.

Profession	Loyalty
Labors and Servants	1
Students	2
Retired\Old Age Citizens	3
Government Servants	4
Business Man/ Self Employed Person	5
Entrepreneur and Industrialists	6

Account Type (Loyalty 1 to 3)

The customer with a Saving Account is more loyal to the bank than those with Current Account.

Profession	Loyalty
Current Account	1
Saving Account	2
Joint Saving Account	3

4.5. Calculation of the Data Dissimilarity

Clustering is the process of grouping the data objects or items into classes or clusters, so that objects within a cluster or class have high similarity in comparison to one another but are very dissimilar to objects in other clusters. An effective clustering technique tries to minimize the intra-cluster dissimilarity among the data items and maximize the inter-cluster similarity. Dissimilarities of the data objects are assessed based on the attribute values describing the objects.

Often, distance measures are used. Clustering has its roots in many areas, including data mining, statistics, biology, and machine learning.

Suppose that a data set to be clustered contains n objects, which may represent persons, houses, documents, countries, and so on. Main memory-based clustering algorithms typically operate on either of the following two data structures.

a) Data matrix (or object-by-variable structure)

This represents ***n objects***, such as persons, with ***p variables*** (also called measurements or attributes), such as age, height, weight, gender, and so on. The structure is in the form of a relational table, or ***n-by-p matrix*** (n objects x p variables):

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$

Figure 4.2: Data matrix (object-by-variable structure)

b) Dissimilarity matrix (or object-by-object structure):

This stores a collection of proximities that are available for all pairs of n objects. It is often represented by an n-by-n table. An example of the Dissimilarity matrix is given below, Where d(i, j) is the measured difference or dissimilarity between objects i and j. In general, d(i, j) is a nonnegative number that is close to 0 when objects i and j are highly similar or "near" each other, and becomes larger the more they differ. Since d(i, j) =d(j, i), and d(i, i)=0, the above matrix shows these entries.

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix}$$

Figure 6: ***Dissimilarity Matrix*** (object-by-object structure)

The rows and columns of the ***Data Matrix*** represent different entities, while those of the ***Dissimilarity Matrix*** represent the same entity. Thus, the data matrix is often called a two-mode matrix, whereas the dissimilarity matrix is called a one-mode matrix. Many clustering algorithms operate on a dissimilarity matrix. If the data are presented in the form of a *Data Matrix*, it can first be transformed into a Dissimilarity Matrix before applying such clustering algorithms.

The data matrix for the bank customers in the target dataset has been developed by combining the credential of the customers after suppression of the sensitive attributes. Only seven attributes (Annual Income, Subscription Period, Profession, Age, Education, Account Type and Marital Status) of a customer's record were included in the data matrix. This data matrix is organized in the form a table (or a spread sheet) to produce the hard copy of the data matrix, while the data matrix exists in the form of a relation of the database (Bank Database or bankdb). The following figure-4 shows a sample of the customer's data matrix. A single row of the data matrix represents record of an individual customer; while the each column of the data represents a domain of attributes from which each record takes a single value.

The next figure (figure-5) shows another version of the data matrix that was developed after applying the *Linear Data Transformation* to preserve privacy of the data. The transformation have been done according the Linear Transformation Table (given in figure-4) to Encrypt the Customer's Data. It is easy to understand the level of privacy achieved through the transformation by comparing the two data matrices. Clearly the encrypted data matrix gives apparently no information about identity of an individual record yet it is providing the input to the clustering process. It is worth to mention that the Customer's ID (C-ID) is also an encoded/ transformed form of the original Customer's Unique ID. The Customer's ID has been kept in the numeric format because of its extensive use in the database as the primary key of an individual record, to carry out different types of processing on data.

Table 4.2: A Sample of the Customer's Data Matrix

The Customer's Data							
C-ID	**Annual Income**	**Subscription Period**	**Profession**	**Age**	**Education**	**Account Type**	**Marital Status**
0001	85501	Between 5 to 8 years	Government Servant	Between 50 to 60 Years	Masters	Saving	Married
0002	97638	Less than 2 years	Business Man / Self-employed	Between 18 to 25 Years	Graduation OR Intermediate	Saving	Married
0003	87800	Between 5 to 8 years	Government Servant	Between 25 to 35 Years	Matriculate OR Below	Joint Saving Account	Single
0004	77436	Between 2 to 5 years	Labor/Servant	Between 25 to 35 Years	Masters	Saving	Single
0005	95558	Less than 2 years	Retired/Old-Age Citizen	Between 25 to 35 Years	M.Phil OR Phd	Joint Saving Account	Single
0006	99027	Less than 2 years	Business Man / Self-employed	Between 35 to 50 Years	M.Phil OR Phd	Joint Saving Account	Married
0007	92784	Between 2 to 5 years	Entrepreneur/Industrialist	Between 18 to 25 Years	Masters	Saving	Single
0008	97656	Between 2 to 5 years	Government Servant	Between 18 to 25 Years	Masters	Saving	Single
0009	89548	Between 2 to 5 years	Government Servant	Between 50 to 60 Years	Matriculate OR Below	Joint Saving Account	Single
0010	93274	Less than 2 years	Government Servant	Between 50 to 60 Years	Graduation OR Intermediate	Joint Saving Account	Single
0011	77043	Between 5 to 8 years	Retired/Old-Age Citizen	Between 18 to 25 Years	Masters	Current	Married
0012	78992	Between 5 to 8 years	Government Servant	Between 35 to 50 Years	Masters	Joint Saving Account	Single
0013	78491	Between 2 to 5 years	Government Servant	Above than 60 Years	Masters	Current	Single
0014	96877	Less than 2 years	Entrepreneur/Industrialist	Between 18 to 25 Years	Graduation OR Intermediate	Current	Single
0015	76154	Between 5 to 8 years	Student	Between 18 to 25 Years	Graduation OR Intermediate	Joint Saving Account	Single
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
4998	510138	Above than 8 years	Labor/Servant	Between 50 to 60 Years	Graduation OR Intermediate	Saving	Married
4999	532637	Less than 2 years	Business Man / Self-employed	Between 18 to 25 Years	Graduation OR Intermediate	Joint Saving Account	Single
5000	680552	Between 5 to 8 years	Government Servant	Above than 60 Years	Graduation OR Intermediate	Saving	Divorced

Table 4.3: A sample of the Encrypted Data Matrix of Customer's Loyalty Data

Encrypted Customer's Data							
Customer's ID	Annual Income	Subscription Period	Profession	Age	Education	Account Type	Marital Status
0001	5	3	4	4	3	2	2
0002	4	1	5	1	2	2	2
0003	2	3	4	2	1	3	1
0004	6	2	1	2	3	2	1
0005	3	1	3	2	4	3	1
0006	3	1	5	3	4	3	2
0007	6	2	6	1	3	2	1
0008	6	2	4	1	3	2	1
0009	3	2	4	4	1	3	1
0010	6	1	4	4	2	3	1
0011	1	3	3	1	3	1	2
0012	2	3	4	3	3	3	1
0013	2	2	4	5	3	1	1
0014	6	1	6	1	2	1	1
0015	1	3	2	1	2	3	1
0016	4	1	2	2	2	2	1
0017	1	1	6	2	4	3	1
0018	6	1	2	3	2	2	2
0019	6	1	3	3	3	3	1
0020	6	4	3	3	3	1	2
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
4995	1	2	2	1	4	1	2
4996	3	1	2	4	4	1	1
4997	2	4	4	5	1	1	2
4998	2	4	1	4	2	2	2
4999	4	1	5	1	2	3	1
5000	6	3	4	5	2	2	2

4.6. The Data Dissimilarity Measures

The section describes the distance measures that are commonly used for computing the dissimilarity of data objects. These measures include the Euclidean, Manhattan, and Minkowski distances. These measures are usually applied on the Interval-scaled variables. These variables are continuous measurements of a roughly linear scale. Typical examples include Age and Height, Annual Income etc.

It is worth to mention that the unit of measurement to measure an attribute can affect the clustering analysis. For example, changing measurement unit from hundred to thousand for annual income or meters to inches for height, may lead to a very different clustering structure and output clusters. In general, expressing a variable in smaller units will lead to a larger range for that variable, and thus a larger effect on the resulting clustering structure. To help avoid dependence on the choice of measurement units, the data should be standardized. Standardizing measurements attempts to give all variables an equal weight. This is particularly useful when given no prior knowledge of the data. Sometime a user may intentionally want to give more weight to a certain set of attributes than to others. For example, when clustering *Loyal Bank Customers*, it seems better to give more weight to the average number of monthly transactions than to the amount of money transacted monthly. To standardize measurements, it is better to convert the original measurements to a unit with fewer variables.

After standardization or without standardization if not necessarily required, the dissimilarity matrix (or similarity matrix) is computed based on the distance between each pair of objects. Some of the popular distance measures to build the dissimilarity matrix are given below.

a. Manhattan distance

The Manhattan distance, d1, between two vectors in an n-dimensional real vector space with fixed Cartesian coordinate system is the sum of the lengths of the projections of the line segment between the points onto the coordinate axes. For example, in the plane, the Manhattan distance between (p_1, p_2) and (q_1, q_2) is $| p_1 - q_1 | + | p_2 - q_2 |$. In general we can express it as:

$$d_1(\mathbf{p}, \mathbf{q}) = ||\mathbf{p} - \mathbf{q}||_1 = \sum_{i=1}^{n} |p_i - q_i|,$$

Where

$$\mathbf{p} = (p_1, p_2, \ldots, p_n) \quad \text{and} \quad \mathbf{q} = (q_1, q_2, \ldots, q_n)$$

are two different vectors having n dimensions.

b. Euclidean distance

In mathematics, the Euclidean distance or Euclidean metric is the "ordinary" distance between two points that one would measure with a ruler, which can be proven by repeated application of the Pythagorean Theorem. By using this formula as distance, Euclidean space becomes a metric space. The associated norm is called the Euclidean norm. The Euclidean distance between points P = (p_1, p2, p_3... p_n) and Q = (q_1, q2, q_3... q_n), in Euclidean n-space, is defined as:

$$\sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}.$$

Older literature refers to this metric as Pythagorean metric. The technique has been rediscovered numerous times throughout history, as it is a logical extension of the Pythagorean Theorem.

Both the Euclidean distance and Manhattan distance satisfy the following mathematic requirements of a distance function:

- p(i, j) >= 0 — The Distance is a nonnegative number.
- p(i, i) = 0 — The Distance of an object to itself is 0.
- p(i, j) = p(j, i) — Distance is a symmetric function.
- p(i, j) <= p(i, h) + p(h, j) — Going directly from object **i** to object **j** in space is no more than making a detour over any other object **h** (triangular inequality).

c. Minkowski distance

The Minkowski distance is a metric on Euclidean space which can be considered as a generalization of both the Euclidean distance and the Manhattan distance.

The Minkowski distance of order p between two points:

$$P = (x_1, x_2, \ldots, x_n) \text{ and } Q = (y_1, y_2, \ldots, y_n) \in \mathbb{R}^n$$

is defined as:

$$\left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p}.$$

Minkowski distance is typically used with p being 1 or 2.

In the research work Manhattan Distance measure has been used to compute the dissimilarity; it not only simplified the computation process but also reduced the processing load on the machine. The

dissimilarity measurement of *Manhattan Distance* was done through the Structural Query Language (SQL). Initially a view (Cust_VU) was created to implement the Manhattan Distance Logic. The view could be queried by the following SQL query. It is necessary to mention here that due to the heavy processing load, I have to execute the following code multiple times with usually 100 to 200 customers per execution although a single execution for all the customers is possible in case of availability of a faster data processor. The view was created to obtain the output in the desired format nevertheless one can execute the same code on a relation or table instead of the view. It is also possible to store the calculated Customers Dissimilarity directly into an already existing relation say "DissimilarityMatrix". In such case it is better to write an update SQL statement instead of querying the view. I had to split the data processing and update due to the very long execution time.

The SQL query to implement the Manhattan Distance Measure

```
Create or Replace View CUST_VU
AS
SELECT      C1.Customer_id C1_ID, C2.Customer_id C2_ID,
            ABS( C1.Income - C2.income ) +
            ABS( C1.Sub_Period - C2.Sub_Period )+
            ABS( C1.Profession - C2.Profession ) +
            ABS( C1.Age - C2.Age ) +
            ABS( C1.Education - C2.Education ) +
            ABS( C1.Acc_Type - C2.Acc_Type  ) +
            ABS( C1.Marital_Status - C2.Marital_Status) CLD
FROM        Customers C1,   Customers C2
WHERE       C1.Customer_id   Between  1  AND 5000
AND         C2.Customer_id   Between 1  AND 5000;
```

Querying the View

```
SELECT   C1_ID||','||C2_ID||','||CLD||',' Ref
FROM     Cust_vu
WHERE        C1_ID Between 1 AND 5000
AND          C2_ID Between 1 AND 5000
ORDER By     C1_id, C2_id
```

A sample of the resultant dissimilarity matrix is given in the following table.

Table 4.4: A Sample of the Data Dissimilarity Matrix

The Data Dissimilarity Matrix												
CID	**0001**	**0002**	**0003**	**0004**	**0005**	**0006**	**0007**	**0008**	**0009**	**0010**	.	.
0001	0											
0002	8	0										
0003	9	9	0									
0004	8	10	11	0								
0005	10	8	7	8	0							
0006	8	6	9	12	4	0						
0007	8	6	11	6	10	10	0					
0008	6	6	9	4	8	10	2	0				
0009	7	9	4	11	7	7	11	9	0			
0010	6	8	9	8	8	8	8	6	5	0		
0011	9	9	8	11	9	11	11	9	12	15	0	
0012	6	10	3	10	6	6	10	8	5	8	.	0
0013	7	11	8	11	9	9	11	9	6	9	.	.
0014	11	5	12	9	11	11	3	5	12	7	.	.
0015	12	10	5	10	8	12	12	10	9	12	.	.
0016	9	5	8	5	5	9	9	7	8	7	.	.
0017	13	9	8	13	5	5	9	11	10	11	.	.
0018	7	7	12	5	9	9	9	7	10	5	.	.
0019	7	9	10	5	5	7	7	5	8	3	.	.
0020	5	11	12	7	11	11	9	7	12	9	.	.
0021	9	7	8	5	5	9	7	5	8	11	.	.
0022	5	11	10	5	9	11	9	7	10	9	.	.
0023	3	5	6	7	9	9	7	5	8	7	.	.
0024	5	7	10	7	7	5	7	5	8	3	.	.
0025	12	10	5	10	10	12	14	12	9	14	.	.
0026	9	13	8	15	11	9	11	13	8	11	.	.
0027	8	2	9	8	8	8	4	4	9	6	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.

4.7. Majors steps of the Privacy Preserving Density Based Clustering of Multidimensional Data (PPDB-CMD)

a. Construction of the Data Dissimilarity Matrix (DDM) using the given *Data Matrix.* (In this research work the Encrypted Data Matrix consists of seven attributes of customers of a bank)

b. Finding the Maximal Customer Loyalty Difference (MCLD) in the given Data Matrix to estimate possible parameters for the density based clustering.

c. Counting the Maximum Number of Directly Density Reachable Objects (MDDROs) against all the data items / objects.

d. Setting the suitable parameters for the Density Based Clustering algorithm according to the MDDROs, required minimum density of a cluster and maximum acceptable distance /difference among the data items of a cluster. These parameters are:

- Minimum Points (MinPoints)
- Epsilon Neighborhood (ε-neighborhood)

e. Identification of the CORE Objects from the given dataset according to the clustering parameters (ε-neighborhood and Minimum Points)

f. Identification of the CORE Object which contains MAXIMUM Number of Directly Density Reachable Objects (DDRO) in its ε-neighborhood. This CORE object Is selected as the cluster representative for the first possible cluster.

g. Identification of the Density Reachable Objects (DROs) for the CORE object having MAXIMUM Number of Directly Density Reachable Objects (DDRO) in its ε-neighborhood. These CORE objects are termed as the Core Objects at LEVEL-1.

h. Identification of the CORE Objects those are in ε-neighborhood of all the Core Objects of LEVEL-1. They would be the possible Core Objects at LEVEL-2.

I. Discover the next possible LEVEL of the under developing cluster by repeating the step 'h' for the CORE objects of the latest discovered clustering LEVEL. This process continues till the discovery of all the possible LEVELS of the under developing cluster.

j. Identification of the Core Objects at all the discovered clustering Levels; they all are Density Reachable from each other. These CORE objects collectively form a single Density Based Cluster.

k. Identification of the objects, those are in the ε-neighborhood of any one of the Core Objects at all the discovered LEVELS. They themselves may or may not be the CORE according to the selected clustering parameters (MinPoints and ε-neighborhood). They are the 'Density Connected'.

l. The above mentioned clustering process continuous to discover the next possible cluster in the remaining data objects or items; for the continuation there must exist at least one CORE object. If

no core object exists and the number of remaining data items is less than the minimum acceptable cluster density then these items are declared as the *Outliers*.

m. The e-neighborhood or minimum distance is kept constant throughout the clustering process while the minimum number of points may be reduced to avoid the freezing of the clustering process. Initially it is kept higher enough to discover thickest possible data cluster and later may be gradually reduced to an acceptable level.

n. After discovery of the first cluster, provision of summary of the cluster(s) discovered so for.

o. Include the statistics related to the Outliers (or noisy data objects) in the summary of the clustering process after discovery of all the possible clusters.

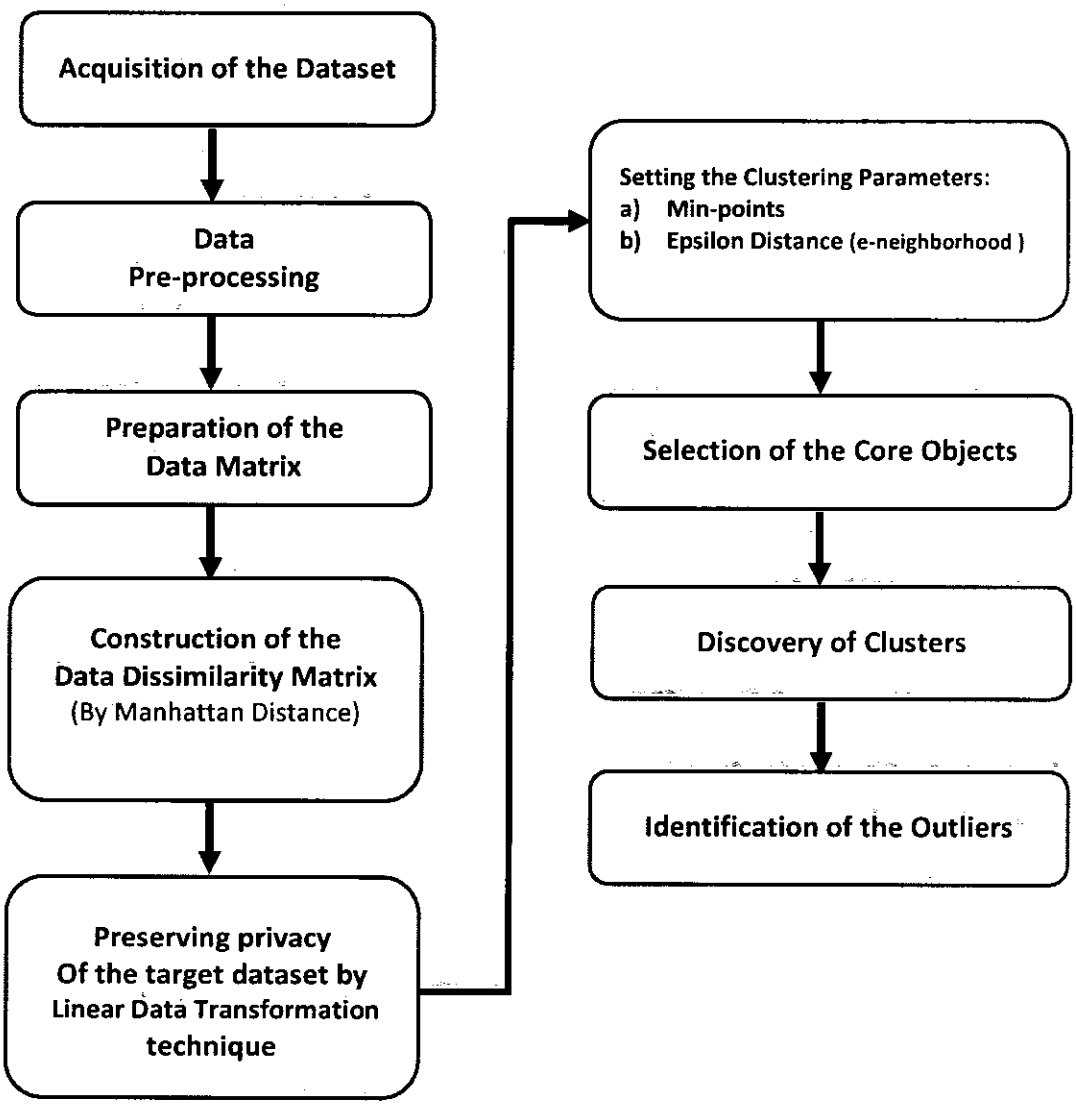


Figure 4.2: Flow Chart of Implementation of the Algorithm

Pseudo code for the algorithm

The proposed Density Based Clustering of Multidimensional Data can be done using the following algorithm:

a) Get the Dataset **D,** minimum number of points **(MinPts)** and the Epsilon Distance **Eps**.

b) Arbitrary select a point ***p,*** from the given dataset ***D***.

c) Retrieve all points those are density-reachable from ***p*** wrt. ***Eps*** and ***MinPts***.

IF $N_{Eps}(p)$ >= MinPts THEN mark ***p*** as a ***Core Point***

ELSE ***p*** is a either a border point, or an Outlier

END IF

d) Repeat steps (b) and (c) for all points in **D**, to discover all ***Core Points (CP)***

e) Arbitrary select a *Core Point* CP_i to discover a cluster C_i

IF no other *Core Point* is at ***Eps*** FROM CP_i

THEN Declare C_i itself is a cluster with CP_i as its cluster representative

ELSE IF CP_i has others *Core Points* i.e. CP_1 *to* CP_N at ***Eps***

THEN FOR CP_1 *to* CP_N *Do*

(1) Make all directly density-reachable points of CP_j as members of the Cluster C_i (j = 1...N)

(2) IF a new *Core Point* CP_x is added as a member of the Cluster C_i x <> (1...N)

THEN Go To (1)

(3) Mark all points of Cluster C_i as *Density Connected* to each other

END IF

END IF

f) Identify the *Core Points* those are not members of any of the discovered cluster(s)

IF *Core Point(s)* are identified

THEN Repeat steps (e) and (f) for them *Core Points*

ELSE End the Clustering process and declare the resultant cluster(s)

END IF

g) Identify the points not belonging to any one of the discovered clusters, as "Noise / Outliers".

The next section describes discovery process of a cluster in step by step fashion. The clustering process discovers seven clusters in total. On completion of the clustering process, the remaining data objects are treated as the *Outliers*. Since the steps to discover a cluster are similar therefore, in order to keep the documentation short, discovery steps of only three clusters are given in the next section.

4.8. Discovering the First Cluster (Cluster1)

Step 1	Identification of the CORE Objects in Cluster1 Table When the **ε-neighborhood = 4 and Minimum Points > = 300**

Update Cluster1
Set Core = 1
Where MDDRO > = 300;
11 rows updated.

Select CID
From Cluster1
Where MDDRO > = 300
Order By CID;
1, 16, 23, 94, 96, 104, 208, 262, 284, 314, 507

Step 2	Identification of the CORE Object in Cluster1 table those contains MAXIMUM Number of Directly Density Reachable Objects (DDRO) in its ε-neighborhood

Select CID, MDDRO From Cluster1
Where MDDRO > = 300
Order By MDDRO DESC;

Customer's ID	Maximum DDRO
C96	368
C23	339
C1	331
C94	328
C314	327
C16	316
C104	313
C262	312
C284	308
C507	307
C208	305

The above table shows that Object with ID = C96, has the largest number of objects in its ε-neighborhood. We also need to find the cluster having the largest number of Density Reachable Objects.

Step 3	Identification of the Density Reachable Objects for every CORE Object in Cluster1 Table

To do the desired calculations RUN the following SQL script.

```
Select CID From Cluster1 Where  CID IN ( Select CID from DM1 Where  C1   < = 4 ) AND Core = 1;
Select CID From Cluster1 Where  CID IN ( Select CID from DM1 Where  C16  < = 4 ) AND Core = 1;
Select CID From Cluster1 Where  CID IN ( Select CID from DM1 Where  C23  < = 4 ) AND Core = 1;
Select CID From Cluster1 Where  CID IN ( Select CID from DM1 Where  C94  < = 4 ) AND Core = 1;
Select CID From Cluster1 Where  CID IN ( Select CID from DM1 Where  C96  < = 4 ) AND Core = 1;
Select CID From Cluster1 Where  CID IN ( Select CID from DM1 Where  C104 < = 4 ) AND Core = 1;
Select CID From Cluster1 Where  CID IN ( Select CID from DM1 Where  C208 < = 4 ) AND Core = 1;
Select CID From Cluster1 Where  CID IN ( Select CID from DM2 Where  C262 < = 4 ) AND Core = 1;
Select CID From Cluster1 Where  CID IN ( Select CID from DM2 Where  C284 < = 4 ) AND Core = 1;
Select CID From Cluster1 Where  CID IN ( Select CID from DM2 Where  C314 < = 4 ) AND Core = 1;
Select CID From Cluster1 Where  CID IN ( Select CID from DM3 Where  C507 < = 4 ) AND Core = 1;
```

The script will result following Output

Core Objects	Link-List of the Density Reachable Objects
C1	{ 23, 94, 284, 314, 507 } → [1, 23, 94, 96, 104, 208, 262, 284, 314, 507]
C16	{ 96, 208 } → [16, 96, 104, 208, 262, 284, 314, 507]
C23	{ 96, 284, 314, 507 } → [→ 1]
C94	{ 507 } → [→ 1]
C96	{ 104, 208, 314, 507 } → [→ 1]
C104	{ 104, 262, 284, 314 } → [→ 1]
C208	{ 262 } → [→ 1]
C262	{ None }→ [→ 1]
C284	{ 314 } → [→ 1]
C314	{ None } → [→ 1]
C507	{ None } → [→ 1]

The above table shows that Object with **Customer's ID = C1**, has the Maximum Number of Density Reachable objects. So it seems justified to select it as the Cluster Representative of a possible cluster in the data.

Step 4	Identification of the CORE Objects in Cluster1 Table those are in ε-neighborhood of the C1. They are the Core Objects at LEVEL-1.

Update Cluster1

Set DR1 = 1

Where CID IN (Select CID from DM1 Where C1 < = 4)

AND Core = 1

AND CID <> 1;

5 rows updated.

Select CID

From Cluster1

Where DR1> = 300

Order By CID;

There following 5 Core Objects are in the in ε-neighborhood of the C1.

C23, C94, C284, C314 and the C507

Step 5	Identification of the CORE Objects in Cluster2 Table, those are in ε-neighborhood of the Core Objects of LEVEL-1. They would be the possible Core Objects at LEVEL-2.

To do the desired calculations RUN the following SQL script.

```
Update Cluster1 Set DR2 = 1 Where CID IN ( Select CID from DM1 Where C23 < = 4 )
AND Core = 1 AND CID NOT IN (1, 23, 94, 284, 314, 507);
Update Cluster1 Set DR2 = 1 Where CID IN ( Select CID from DM1 Where C94 < = 4 )
AND Core = 1 AND CID NOT IN (1, 23, 94, 284, 314, 507);
Update Cluster1 Set DR2 = 1 Where CID IN ( Select CID from DM2 Where C284 < = 4 )
AND Core = 1 AND CID NOT IN (1, 23, 94, 284, 314, 507);
Update Cluster1 Set DR2 = 1 Where CID IN ( Select CID from DM2 Where C314 < = 4 )
AND Core = 1 AND CID NOT IN (1, 23, 94, 284, 314, 507);
Update Cluster1 Set DR2 = 1 Where CID IN ( Select CID from DM3 Where C507 < = 4 )
AND Core = 1 AND CID NOT IN (1, 23, 94, 284, 314, 507);
```

After the UPDATE only ONE new core Object (C96) is found in the ε-neighborhood of the Core Objects of LEVEL-1. So C96 is the Core Object at LEVEL-2

Step 6	Identification of the CORE Objects in Cluster1 Table, those are in ε-neighborhood of the Core Object of LEVEL-2. They would be the possible Core Objects at LEVEL-3.

Update Cluster1
Set DR3 = 1
Where CID IN (Select CID From DM1 Where C96 < = 4)
AND CORE = 1
AND CID NOT IN (1, 23, 94, 96, 284, 314, 507);

2 rows updated.

Only TWO Core Objects (C104, C208) are found.

Step 7	Identification of the CORE Objects in Cluster1 Table, those are in ε-neighborhood of the Core Objects of LEVEL-3. They would be the possible Core Objects at LEVEL-4.
To do the desired calculations RUN the following SQL script.	
Update Cluster1 Set DR4 = 1 Where CID IN (Select CID from DM1 Where C104 < = 4) AND Core = 1 AND CID NOT IN (1, 23, 94, 96, 104, 208, 284, 314, 507); Update Cluster1 Set DR4 = 1 Where CID IN (Select CID from DM1 Where C208 < = 4) AND Core = 1 AND CID NOT IN (1, 23, 94, 96, 104, 208, 284, 314, 507);	

After the UPDATE, No new core Object was found in the ε-neighborhood of the Core Object(s) of LEVEL-4. So the Clustering process for the under developing Cluster cannot proceed further.

Step 8	Identification of the CORE Objects in Cluster1 Table, those are in ε-neighborhood of the Core Objects of LEVEL-5. They would be the possible Core Objects at LEVEL-5.

Update Cluster1 Set DR5 = 1 Where CID IN (Select CID from DM1 Where C262 < = 4)
AND Core = 1
AND CID NOT IN (1, 23, 94, 96, 104, 208, 262, 284, 314, 507);

After the UPDATE, No new core Object was found in the ε-neighborhood of the Core Object(s) of LEVEL-4. So the Clustering process for the under developing Cluster cannot proceed further.

Step 9	Identification of the **Core Objects** at all the Levels; they all are Density Reachable from each other. These CORE objects collectively form a single Density Based Cluster.

Update Cluster1
SET DRO = 1
Where DR1 = 1 OR DR2 = 1 OR DR3 = 1 OR DR4 = 1 OR CID = C1;
10 rows updated.

So the following TEN Core objects are part of the Density Based Cluster.
1, 23, 94, 96, 104, 208, 262, 284, 314, 507

Step 10	Identification of the objects, those are in the ε-neighborhood of any one of the Core Objects at all the discovered LEVELS. These objects themselves may or may not be CORE. They are the 'Density Connected' objects with the object having **Customer ID = C1**. So **C1** is their Cluster Representative.

To do the desired calculations RUN the following SQL script.

```
Update Cluster1 Set DCO1 = 1 Where CID IN ( Select CID from DM1 Where C1 < = 4 );
Update Cluster1 Set DCO1 = 1 Where CID IN ( Select CID from DM1 Where C23 < = 4 );
Update Cluster1 Set DCO1 = 1 Where CID IN ( Select CID from DM1 Where C94 < = 4 );
Update Cluster1 Set DCO1 = 1 Where CID IN ( Select CID from DM1 Where C96 < = 4 );
Update Cluster1 Set DCO1 = 1 Where CID IN ( Select CID from DM1 Where C104 < = 4 );
Update Cluster1 Set DCO1 = 1 Where CID IN ( Select CID from DM1 Where C208 < = 4 );
Update Cluster1 Set DCO1 = 1 Where CID IN ( Select CID from DM2 Where C262 < = 4 );
Update Cluster1 Set DCO1 = 1 Where CID IN ( Select CID from DM2 Where C284 < = 4 );
Update Cluster1 Set DCO1 = 1 Where CID IN ( Select CID from DM2 Where C314 < = 4 );
Update Cluster1 Set DCO1 = 1 Where CID IN ( Select CID from DM3 Where C507 < = 4 );
```

Select Count(CID) From Cluster1 Where DCO1 = 1;

COUNT(CID)

1809

After the UPDATE there were 1809 objects out of the 5000, those are Density Connected to C1. They collectively form a Density Based Cluster with C1 as the Cluster Representative.

4.9. Discovering the Second Cluster (Cluster2)

Before moving further, it seems logical to summarize the so far Clustered Data.

Clustering Summary
Summary of the Clustering process that have been completed
So far, only ONE Clusters has been identified in the Data • **1809** Objects (Customers) with 'C1' as the Cluster Representative when **ε**-Neighborhood = 4 And Minimum Points > = 300 **Overall 1809** objects [Table Cluster1] have been Clustered Out of the 5000 objects.

Step 1	Identification of the CORE Objects in Cluster2 Table, When the **ε**-Neighborhood = 4 and Minimum Points > = 250

```
Update Cluster2
Set Core = 1
Where   MDDRO > = 250;
AND     Clustered  = 0;
```

12 rows updated.

```
Select CID
From Cluster2
Where MDDRO > = 250
Order By CID;
```

```
 CID
-----
  12
  16
  21
  22
  47
  55
  90
 103
 175
 187
 312
 423
```

12 rows selected

Step 2	Identification of the CORE Object in Cluster2 Table that contains MAXIMUM Number of DDRO in its ε-neighborhood

Update Cluster2
Set Core = 1
Where MDDRO > = 250 AND Clustered = 0;

Customer's ID	Maximum DDRO
C16	316
C12	290
C22	279
C103	272
C175	266
C312	265
C423	261
C55	261
C187	258
C21	257
C47	255
C90	252

The above table shows that Object with **Customer's ID = C16**, has the largest number of objects in its ε-neighborhood. We also need to find the Core Object that has maximum number of Density Reachable objects.

Step 3	Identification of the Density Reachable Objects for every CORE Object in Cluster2 Table.

To do the desired calculations RUN the following SQL script.

```
Select CID From Cluster2 Where CID IN ( Select CID from DM1 Where C12 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM1 Where C16 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM1 Where C21 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM1 Where C22 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM1 Where C47 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM1 Where C55 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM1 Where C90 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM1 Where C103 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM1 Where C175 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM1 Where C187 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM2 Where C312 < = 4 ) AND Core = 1;
Select CID From Cluster2 Where CID IN ( Select CID from DM2 Where C423 < = 4 ) AND Core = 1;
```

The script will result following Output

Core Objects	Density Reachable Objects
C12	{ 47, 103, 175, 187 } → [12, 47, 103, 175, 187, 312, 423]
C16	{ 21, 55 } → [16, 21, 55, 187, 312]
C21	{ 187 } → [→ 12]
C22	{ none } → [22]
C47	{ 175 } → [→ 12]
C55	{ 187, 312} → [→ 12]
C90	{ none } → [90]
C103	{ 175, 312} → [→ 12]
C175	{ 423 } → [→ 12]
C187	{ none } → [→ 12]
C312	{ none } → [→ 12]
C423	{ none } → [→ 12]

The above table shows that Object with **Customer's ID = C12**, has the Maximum Number of Density Reachable objects. So it seems justified to select it as the Cluster Representative of a possible cluster in the data.

Step 4	Identification of the CORE Objects in Cluster1 Table those are in ε-neighborhood of the C12. They are the Core Objects at LEVEL-1.

```
Update Cluster2
Set DR1 = 1
  Where  CID  IN  ( Select CID from DM1 Where  C12 < = 4 )
AND Core = 1    AND CID <> 12;
```

4 rows updated.

```
Select CID
From Cluster1
Where  DR1  =  1
Order By CID;
```

There following 4 Core Objects are in the in ε-neighborhood of the C12.

C47, C103, C187 and C175

Step 5	Identification of the CORE Objects in Cluster2 Table, those are in ε-neighborhood of the Core Objects of LEVEL-1. They would be the possible Core Objects at LEVEL-2.

To do the desired calculations RUN the following SQL script.

```
Update Cluster2  Set DR2 = 1   Where  CID IN ( Select CID from DM1 Where  C47  < = 4 )
AND     Core = 1    AND     CID NOT IN (12, 47, 103, 187, 175);

Update Cluster2  Set DR2 = 1   Where  CID IN ( Select CID from DM1 Where  C103 < = 4 )
AND     Core = 1    AND     CID NOT IN (12, 47, 103, 187, 175);

Update Cluster2  Set DR2 = 1   Where  CID IN ( Select CID from DM1 Where  C187 < = 4 )
AND     Core = 1    AND     CID NOT IN (12, 47, 103, 187, 175);

Update Cluster2  Set DR2 = 1   Where  CID IN ( Select CID from DM1 Where  C175 < = 4 )
AND     Core = 1    AND     CID NOT IN (12, 47, 103, 187, 175);
```

After the update TWO new core Objects (C312, C423) are found in the ε-neighborhood of the Core Objects of LEVEL-1. They are the Core Objects at LEVEL-2.

Step 6	Identification of the CORE Objects in Cluster2 Table, those are in ε-neighborhood of the Core Objects of LEVEL-2. They would be the possible Core Objects at LEVEL-3.

```
Update Cluster2   Set DR3 = 1  Where  CID IN ( Select CID from DM2 Where  C312 < = 4 )
AND  CORE = 1
AND  CID  NOT  IN ( 12 , 47, 103, 187, 175, 312, 423 );
```

```
Update Cluster2      Set DR3 = 1  Where  CID IN ( Select CID from DM2 Where  C423 < = 4 )
AND      CORE = 1
AND CID NOT IN ( 12 , 47, 103, 187, 175, 312, 423 );
```

After the UPDATE, No new core Object was found in the ε-neighborhood of the Core Object(s) of LEVEL-2. So the Clustering process for the under developing Cluster cannot proceed further

Step 7	Identification of all the Core Objects at the all Levels. They all are Density Reachable from each other. These CORE objects collectively form a single Density Based Cluster.

```
Update Cluster2
SET DRO12  = 1
Where  DR1 = 1   OR    DR2 = 1   OR    CID = C12;
```

7 rows updated.

```
Select CID
From Cluster2
Where DRO12 = 1
Order by CID;
```

CID

12
47
103
175
187
312
423

So the SEVEN Core objects are part of the Density Based Cluster.

Step 8	Identification of the objects those are in the ε-neighborhood of any one of the Core Objects of any one of the Levels. These objects may or may not be CORE themselves. These Objects are termed as 'Density Connected' objects with the object having **Customer ID = C12**. The **C12** is the starting Core Object of the discovered cluster.

To do the desired calculations RUN the following SQL script.

```
Update Cluster2  Set DCO12 = 1
Where  CID IN ( Select CID from DM1 Where  C12  < = 4 ) AND Clustered  = 0 ;
Update Cluster2  Set DCO12 = 1
Where  CID IN ( Select CID from DM1 Where  C47  < = 4 ) AND Clustered  = 0 ;
Update Cluster2  Set DCO12 = 1
Where  CID IN ( Select CID from DM1 Where  C103 < = 4 ) AND Clustered  = 0 ;
Update Cluster2  Set DCO12 = 1
Where  CID IN ( Select CID from DM1 Where  C175 < = 4 ) AND Clustered  = 0 ;
Update Cluster2  Set DCO12 = 1
Where  CID IN ( Select CID from DM1 Where  C187 < = 4 ) AND Clustered  = 0 ;
Update Cluster2  Set DCO12 = 1
 Where  CID IN ( Select CID from DM2 Where  C312 < = 4 ) AND Clustered  = 0 ;
Update Cluster2  Set DCO12 = 1
Where  CID IN ( Select CID from DM2 Where  C423 < = 4 ) AND Clustered  = 0 ;
```

Select Count (CID) From Cluster2 Where DCO12 = 1;

```
COUNT(CID)
--------------
  647
```

After the UPDATE there were **647** objects out of the 5000, those are Density Connected to C12. They collectively form a Density Based Cluster with C12 as the Cluster Representative.

The same steps are to be repeated to discover the next cluster. To keep the documentation abridged, we skip to put the step for upcoming cluster up to the last cluster. In total seven clusters are found in the dataset. The implementation steps for the seventh cluster are given on the following pages.

4.10. Discovering the SEVENTH Cluster (Cluster7)

Before moving further, it seems logical to summarize the so far Clustered Data

Clustering Summary

Summary of the Clustering process that have been completed

So far, only ONE Clusters has been identified In the Data

- **1809 Objects** (Customers) with '**C1**' as the Cluster Representative when
  Ɛ-Neighborhood = 4 And Minimum Points > = 300
- **647 Objects** (Customers) with '**C12**' as the Cluster Representative
  Ɛ-Neighborhood = 4 And Minimum Points > = 250
- **487 Objects** (Customers) with '**C16**' as the Cluster Representative
  Ɛ-Neighborhood = 4 And Minimum Points > = 225
- **389 Objects** (Customers) with '**C13**' as the Cluster Representative
  Ɛ-Neighborhood = 4 And Minimum Points > = 200
- **744 Objects** (Customers) with '**C3**' as the Cluster Representative
  Ɛ-Neighborhood = 4 And Minimum Points > = 175
- **374 Objects** (Customers) with '**C20**' as the Cluster Representative
  Ɛ-Neighborhood = 4 And Minimum Points > = 150

Overall **4450** = 1809 [TABLE Cluster1] + 647 [TABLE Cluster2] + 487 [TABLE Cluster3] +
389 [TABLE Cluster4] + 374 [TABLE Cluster2]
Objects have been Clustered Out of the 5000 objects.

Step 0	Identification of the Objects those are yet to be Clustered in Cluster7 table.

```
Update Cluster7
Set Clustered = 1
Where CID IN ( Select CID From Cluster7 where Clustered = 1 OR DC20 = 1);
```

4450 rows updated.

Step 1	Identification of the CORE Objects in Cluster7 Table When the ε-Neighborhood = 4 and Minimum Points > = 175

Update Cluster7

Set Core = 1 Where MDDRO > = 120; AND Clustered = 0;

111 rows updated.

Select CID From Cluster7 Where MDDRO > = 150 Order By CID;

2, 4, 5, 6, 7, 8, 10, 11, 15, 19, 24, 25, 26, 27, 28, 29, 33, 35, 38, 40, 42, 45, 46, 49, 51, 52, 54, 56, 61, 62, 63, 64, 67, 77, 90, 97, 98, 110, 125, 131, 134, 135, 139, 140, 145, 152, 153, 158, 165, 166, 167, 171, 174, 177, 184, 198, 199, 200, 201, 205, 210, 213, 227, 230, 240, 241, 245, 249, 271, 274, 275, 287, 297, 298, 301, 324, 331, 339, 354, 364, 375, 393, 420, 435, 456, 468 470, 492, 506, 525, 538, 554, 624, 644, 692, 712, 727, 739, 861, 955, 975, 997, 1066, 1104, 1233, 1319, 1386, 1429, 1494 1716, 1894

111 rows selected

Step 2	Identification of the CORE Object in Cluster7 Table, that contains MAXIMUM Number of DDRO in its ε-neighborhood.

Select CID, MDDRO From Cluster7

Where MDDRO > = 120 AND Clustered = 0 Order By MDDRO DESC;

The SQL query selects following 65 records

	CID	MDDRO	CID	MDDRO	CID	MDDRO
1	90	252	52	156	506	131
2	54	223	324	153	240	131
3	77	222	125	151	975	130
4	29	219	171	150	227	130
5	46	219	140	150	205	130
6	5	210	177	148	1386	130
7	19	209	492	148	62	130
8	2	209	213	147	184	129
9	165	207	297	147	7	129
10	97	204	167	145	166	128
11	6	202	49	145	275	127
12	4	200	25	145	538	127
13	27	195	64	144	727	127
14	210	194	63	144	393	127
15	8	194	110	144	1494	126
16	67	190	298	144	199	125
17	51	190	435	143	1319	125
18	24	184	301	143	245	124
19	15	180	375	142	692	124
20	249	179	997	142	230	124
21	35	178	712	142	1066	124
22	10	177	287	141	644	124
23	174	177	271	141	274	124
24	145	176	364	141	354	123
25	152	174	40	141	1429	122
26	56	173	955	140	131	122
27	42	172	198	139	624	121
28	241	170	468	138	506	131
29	456	169	28	137		
30	38	169	525	137		
31	45	166	201	136		
32	134	164	135	136		
33	200	160	139	133		
34	861	160	331	133		
35	153	159	61	131		
36	11	159	1233	131		
37	98	158	33	131		
38	339	158	470	131		

The above table shows that Object with **Customer's ID = C90**, has the largest number of objects in its ε-neighborhood. We also need to find the cluster having the largest number of Density Based Objects.

Step 3	Identification of the Density Reachable Objects for every CORE Object in Cluster7 Table

To do the desired calculations RUN the following SQL script.

```
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C2 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C4 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C5 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C6 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C7 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C8 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C10 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C11 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C15 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C19 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C24 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C25 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C26 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C27 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C28 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C29 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C33 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C35 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C38 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C40 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C42 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C45 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C46 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C49 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C51 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C52 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C54 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C56 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C61 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C62 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C63 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C64 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C67 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C77 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C90 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C97 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C98 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C110 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C125 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C131 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C134 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C135 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C139 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C140 < = 4 ) AND Core = 1 AND Clustered = 0 ;
```

Step 3	Continued

```
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C145 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C152 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C153 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C158 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C165 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C166 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C167 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C171 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C174 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C177 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C184 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C198 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C199 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C200 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C201 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C205 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C210 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C213 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C227 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C230 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C240 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C241 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C245 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM1 Where C249 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C271 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C274 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C275 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C287 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C297 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C298 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C301 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C324 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C331 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C339 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C354 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C364 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C375 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C393 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C420 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C435 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C456 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C468 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C470 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM2 Where C492 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM3 Where C506 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM3 Where C525 < = 4 ) AND Core = 1 AND Clustered = 0 ;
```

Step 3	Continued

```
Select CID From Cluster7 Where CID IN ( Select CID from DM3 Where C538 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM3 Where C554 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM3 Where C624 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM3 Where C644 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM3 Where C692 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM3 Where C712 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM3 Where C727 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM3 Where C739 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM4 Where C861 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM4 Where C955 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM4 Where C975 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM4 Where C997 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM5 Where C1096 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM5 Where C1204 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM5 Where C1233 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM6 Where C1319 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM6 Where C1386 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM6 Where C1429 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM6 Where C1494 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM7 Where C1716 < = 4 ) AND Core = 1 AND Clustered = 0 ;
Select CID From Cluster7 Where CID IN ( Select CID from DM8 Where C1894 < = 4 ) AND Core = 1 AND Clustered = 0 ;
```

The script will result following Output:

The script will result following Output	
Core Objects	**Density Reachable Objects**
2	{ 27, 35, 167, 199, 213, 271, 375 } → [2, 27, 35, 77, 134, 135, 165, 167, 177, 199, 213, 271, 301, 324, 364, 375, 435, 470, 506, 538, 955, 997, 1233, 1386, 1494]
4	{ 8, 90 } → [8, 27, 42, 77, 90, 134, 135, 167, 177, 199, 271, 301, 364, 375, 435, 470, 506, 538, 955, 997, 1233, 1386, 1494]
5	{ 6, 135 } → [5, 6, 135, 177, 199, 271, 538, 1386, 1494]
6	{ 135, 177, 538 } → [→ 5]
7	{ 8, 27, 64, 167, 199, 271, 331, 506 } → [7, 8, 27, 42, 64, 77, 134, 135, 167, 177, 199, 271, 301, 331, 364, 375, 435, 470, 506, 538, 955, 997, 1233, 1386, 1494]
8	{ 27, 42, 167, 271 } → [→ 4]
10	{ 19, 24, 42, 166, 241 } → [10, 19, 24, 42, 77, 134, 135, 166, 167, 177, 199, 205, 227, 241, 271, 298, 301, 324, 364, 435, 470, 506, 538, 712, 955, 997, 1233, 1319, 1386, 1494]
11	{ 46, 727 } → [11, 46, 727, 861, 955, 1233]
15	{ 25, 38 } → [15, 25, 38, 51]
19	{ 24, 42, 77 } → [→ 10]
24	{ 42, 77, 298 } → [→ 10]
25	{ 38 } → [→ 15]
27	{ 77, 167, 199, 271, 364, 375 } → [→ 2]
28	{ 61, 301 } → [28, 61, 134, 135, 177, 199, 271, 301, 538, 727, 955, 1233, 1386, 1494]
29	{ 145, 152 } → [29, 145, 152, 184, 339]
33	{ 184, 275 } → [33, 184, 275, 339]
35	{ 165, 375 } → [→ 2]
38	{ 51 } → [→ 15]
40	{ 125, 861 } → [40, 125, 139, 140, 171, 198, 227, 241, 249, 298, 324, 364, 435, 456, 470, 492, 506, 712, 861, 955, 975, 997, 1233, 1319, 1386, 1494]
42	{ 77, 167, 271 } → [→ 4]
45	{ 63, 275 } → [45, 63, 275, 339]
46	{ 861 } → [→ 11]
49	{ 171, 298, 324, 506, 997 } → [49, 171, 298, 324, 364, 435, 470, 506, 955, 997, 1233, 1494]
51	{ } → [→ 15]
52	{ 125, 205, 213, 227, 324, 1494 } → [52, 125, 139, 140, 171, 198, 205, 213, 227, 241, 249, 298, 324, 364, 435, 456, 492, 506, 712, 861, 955, 975, 997, 1233, 1319, 1386, 1494]
54	{ 56, 97, 125, 174, 213, 324, 975, 1319 } → [54, 56, 97, 125, 139, 140, 171, 174, 177, 198, 213, 227, 241, 324, 364, 249, 298, 324, 364, 435, 456, 470, 492, 506, 538, 712, 861, 955, 975, 997, 1233, 1319, 1386, 1494]
56	{ 97, 174, 198, 861, 1386 } → [→ 54]
61	{ 134, 135, 199, 301, 727 } → [→ 28]
62	{ 98, 110, 140, 712 } → [62, 98, 110, 140, 171, 198, 227, 241, 298, 324, 364, 435, 470, 506, 712, 955, 997, 1233, 1319, 1494]
63	{ 275 } → [→ 45]
64	{ 331, 506, 997 } → [→ 7]
67	{ 90 } → [67, 90]
77	{ 134, 167, 271, 301, 364 } → [→ 2]
90	{ } → [→ 4]
97	{ 174, 213, 324, 435, 861 } → [→ 54]

The script will result following Output (Continued)	
Core Objects	**Density Reachable Objects**
110	{ 140, 198, 712, 1319 } → [→ 62]
125	{ 139, 140, 171, 249, 456, 492, 975, 1319 } → [→ 40]
134	{ 135, 177, 199, 301, 955, 1233 } → [→ 2]
135	{ 177, 199 } → [→ 2]
139	{ 140, 227, 712, 975, 1319, 1386, 1494 } → [→ 40]
140	{ 171, 198, 227, 712 } → [→ 40]
145	{ 152, 184, 339 } → [→ 29]
152	{ } → [→ 29]
153	{ 201 } → [153]
165	{ } → [→ 2]
166	{ 205, 538 } → [→ 10]
167	{ 199, 271 } → [→ 2]
171	{ 298, 997 } → [→ 40]
174	{ 177, 213, 324, 364, 435 } → [→ 2]
177	{ 538, 1386 } → [→ 2]
184	{ 339 } → [→ 29]
198	{ 227 } → [→ 40]
199	{ 271 } → [→ 2]
200	{ 240, 339 } → [200, 240, 339, 393, 468, 525]
201	{ } → [201]
205	{ 227, 1386, 1494 } → [→ 10]
210	{ 470, 727, 861, 955, 1233 } → [→ 10]
213	{ 324, 364, 435, 506 } → [→ 2]
227	{ 241, 324, 364, 712, 1494 } → [→ 10]
240	{ 393, 525 } → [→ 200]
241	{ 712 } → [→ 10]
249	{ 456, 492, 861, 975, 1386, 1494 } → [→ 40]
271	{ } → [→ 2]
275	{ 339 } → [→ 33]
287	{ } → [→ 10]
297	{ } → [→ 10]
298	{ 324 } → [→ 10]
301	{ } → [→ 2]
324	{ 364, 506, 997 } → [→ 2]
331	{ 364, 506 } → [→ 7]
339	{ } → [→ 29]
364	{ 435, 506 } → [→ 2]
375	{ } → [→ 2]
393	{ 468, 525 } → [→ 200]
435	{ 470, 955, 1233 } → [→ 2]

The script will result following Output (Continued)	
Core Objects	**Density Reachable Objects**
435	{ 470, 955, 1233 } → [→ 2]
456	{ 492, 861, 975, 1386, 1494 } → [→ 40]
468	{ 525 } → [→ 200]
470	{ } → [→ 2]
492	{ 861, 975 } → [→ 40]
506	{ 997 } → [→ 2]
525	{ } → [→ 200]
538	{ 1386 } → [→ 2]
712	{ 1319, 1494 } → [→ 10]
727	{ } → [→ 11]
861	{ 955, 1233 } → [→ 11]
955	{ 1233 } → [→ 2]
975	{ 1319, 1386, 1494 } → [→ 40]
997	{ 1494 } → [→ 2]
1233	{ } → [→ 2]
1319	{ } → [→ 10]
1386	{ 1494 } → [→ 2]
1494	{ } → [→ 2]

The above table shows that Object with **Customer's ID = C54**, has the Maximum Number of Density Reachable objects. So it seems justified to select it as the Cluster Representative of a possible cluster in the data.

Step 4	Identification of the CORE Objects in Cluster7 Table, those are in ε-neighborhood of the C54. They are the Core Objects at LEVEL-1

```
Update Cluster7
Set     DR1 = 1  Where  CID      IN (Select CID from DM1 Where C54 < = 4 )
 AND    Core = 1  AND    CID <> 54;
```

9 rows updated.

```
Select CID
From    Cluster7  Where  DR1  = 1
Order    By  CID;
```

56, 97, 125, 174, 213, 324, 975, 1319, 1894

So the NINE Core Objects are in the in ε-neighborhood of the customer with ID = C54.

Step 5	Identification of the CORE Objects in Cluster7Table, those are in ε-neighborhood of the C1. They are the Core Objects at LEVEL-2

To do the desired calculations RUN the following SQL script.

```
Update Cluster7  Set DR2 = 1   Where  CID IN ( Select CID from DM1 Where  C56 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1);

Update Cluster7  Set DR2 = 1   Where  CID IN ( Select CID from DM1 Where  C97 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1);

Update Cluster7  Set DR2 = 1   Where  CID IN ( Select CID from DM1 Where  C125 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1);

Update Cluster7  Set DR2 = 1   Where  CID IN ( Select CID from DM1 Where  C174 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1);

Update Cluster7  Set DR2 = 1   Where  CID IN ( Select CID from DM1 Where  C213 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1);

Update Cluster7  Set DR2 = 1   Where  CID IN ( Select CID from DM2 Where  C324 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1);

Update Cluster7  Set DR2 = 1   Where  CID IN ( Select CID from DM4 Where  C975 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1);

Update Cluster7  Set DR2 = 1   Where  CID IN ( Select CID from DM6 Where  1319 < = 4 )
AND Core = 1;

Update Cluster7  Set DR2 = 1   Where  CID IN ( Select CID from DM8 Where  1894 < = 4 )
AND Core = 1;
```

17 (distinct) rows updated

Select CID From Cluster7 Where DR2=1 Order by CID;

139, 140, 171, 177, 198, 249, 364, 435, 456, 492, 506, 739, 861, 997, 1386, 1429, 1494

After the update SEVENTEEN new core Objects are found in the ε-neighborhood of the Core Objects of LEVEL-1. They are the Core Objects at LEVEL-2.

Step 6	Identification of the CORE Objects in Cluster7 Table, those are in ε-neighborhood of the Core Objects of LEVEL-2. They are the possible Core Objects at LEVEL-3.

To do the desired calculations RUN the following SQL script.

```
Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM1 Where C139 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM1 Where C140 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM1 Where C171 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM1 Where C177 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM1 Where C198 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster 7 Set DR3 = 1 Where CID IN ( Select CID from DM1 Where C249 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM2 Where C364 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM2 Where C435 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM2 Where C456 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM2 Where C492 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM3 Where C506 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM3 Where C739 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM4 Where C861 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM4 Where C997 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM6 Where C1386 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM6 Where C1429 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);

Update Cluster7 Set DR3 = 1 Where CID IN ( Select CID from DM6 Where C1494 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1);
```

10 (distinct) rows updated

Select CID From Cluster7 Where DR3=1 Order by CID;

```
 CID
------
 227
 274
 298
 470
 538
 712
 955
1066
1233
1716
```

After the UPDATE, THIRTEEN new core Objects were found in the ε-neighborhood of the Core Object(s) of the LEVEL-2. They are the Core Objects at LEVEL-3

Step 7	Identification of the CORE Objects in Cluster7 Table, those are in ε-neighborhood of the Core Objects of LEVEL-3. They are the Core Objects at LEVEL-4.

To do the desired calculations RUN the following SQL script.

```
Update Cluster7 Set DR4 = 1 Where CID IN ( Select CID from DM1 Where C227 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1);

Update Cluster7 Set DR4 = 1 Where CID IN ( Select CID from DM2 Where C274 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1);

Update Cluster7 Set DR4 = 1 Where CID IN ( Select CID from DM2 Where C298 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1);

Update Cluster7 Set DR4 = 1 Where CID IN ( Select CID from DM2 Where C470 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1);

Update Cluster7 Set DR4 = 1 Where CID IN ( Select CID from DM3 Where C538 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1);

Update Cluster7 Set DR4 = 1 Where CID IN ( Select CID from DM3 Where C712 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1);

Update Cluster7 Set DR4 = 1 Where CID IN ( Select CID from DM4 Where C955 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1);

Update Cluster7 Set DR4 = 1 Where CID IN ( Select CID from DM5 Where C1066 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1);

Update Cluster7 Set DR4 = 1 Where CID IN ( Select CID from DM5 Where C1233 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1);

Update Cluster7 Set DR4 = 1 Where CID IN ( Select CID from DM7 Where C1716 < = 4 )
AND Core = 1 AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1);
```

Select CID From Cluster7 Where DR4=1 Order by CID;

After the UPDATE, TWO new core Objects (241, 624) were found in the ε-neighborhood of the Core Object(s) of LEVEL- 3. They are the Core Object at LEVEL-4

Step 8	Identification of the CORE Objects in Cluster7 Table, those are in ε-neighborhood of the Core Objects at Level-4. They would be the possible Core Objects at LEVEL- 5.

To do the desired calculations RUN the following SQL script.

```
Update Cluster7  Set DR5 = 1   Where  CID IN ( Select CID from DM1 Where  C241 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1 OR DR4 = 1);

Update Cluster7  Set DR5 = 1   Where  CID IN ( Select CID from DM3 Where  C624 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1 OR DR4 = 1);
```

Select CID From Cluster7 Where DR5 = 1 Order by CID;

After the UPDATE, TWO new core Objects (554, 644) were found in the ε-neighborhood of the Core Object(s) of LEVEL- 5.

Step 9	Identification of the CORE Objects in Cluster7 Table, those are in ε-neighborhood of the Core Objects of the LEVEL-5. They are the Core Objects at LEVEL- 6.

To do the desired calculations RUN the following SQL script.

```
Update Cluster7  Set DR6 = 1   Where  CID IN ( Select CID from DM3 Where  C554 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1 OR DR4 = 1 OR DR5 = 1);

Update Cluster7  Set DR6 = 1   Where  CID IN ( Select CID from DM3 Where  C644 < = 4 )
AND Core = 1
AND CID NOT IN (Select CID from Cluster7 where DR1 = 1 OR DR2 = 1 OR DR3 = 1 OR DR4 = 1 OR DR5 = 1);
```

After the UPDATE, NO new core Object was found in the ε-neighborhood of the Core Object(s) of LEVEL- 6

Step 10	Identification of all the Core Objects at the all Levels. They all are Density Reachable from each other. These CORE objects collectively form a single Density Based Cluster.

```
Update Cluster7
SET DRO54 = 1
Where
        DR1 = 1   OR    DR2 = 1   OR    DR3 = 1   OR    DR4 = 1   OR    DR5 = 1   OR    CID = 54;
```

41 rows updated.

Step 11	Identification of all the objects that are in the ε-neighborhood of any one of the Core Objects all the discovered Levels. These objects may or may not be CORE themselves. These are the 'Density Connected' objects to the Customer having ID = C54 (The Cluster Starting Core Object)

To do the desired calculations RUN the following SQL script.

```
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C54 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C56 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C97 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C125 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C139 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C140 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C171 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C174 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C177 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C198 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C213 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C227 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C241 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM1 Where C249 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM2 Where C274 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM2 Where C298 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM2 Where C324 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM2 Where C364 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM2 Where C435 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM2 Where C456 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM2 Where C470 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM2 Where C492 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM3 Where C506 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM3 Where C538 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM3 Where C554 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM3 Where C624 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM3 Where C644 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM3 Where C712 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM3 Where C739 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM4 Where C861 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM4 Where C955 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM4 Where C975 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM4 Where C997 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM5 Where C1066 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM5 Where C1233 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM6 Where C1319 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM6 Where C1386 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM6 Where C1429 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM6 Where C1494 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM7 Where C1716 < = 4 ) AND Clustered = 0;
Update Cluster7 Set DCO54 = 1 Where CID IN ( Select CID from DM8 Where C1894 < = 4 ) AND Clustered = 0;
```

Select Count (CID) From Cluster7 Where DCO54 = 1; 306 rows selected

So the Cluster7 has 306 objects those are Density-Connected it. The Customer with **Customer-ID = 54** is the cluster representative.

4.11. Results

The algorithm employs same discovery steps to discover all existing clusters; therefore the previous section describes them only for the first, second and the last cluster. Nevertheless, on successful completion of the clustering process, following clusters were discovered:

a) Cluster 1

Cluster objects, those are in the ε-neighborhood of any one of the Core Objects at all the discovered LEVELS. These objects themselves may or may not be CORE. They are the 'Density Connected' objects with the core object having **Customer ID = C1**.

Cluster 1	
Cluster Representative	Data Object (Customer) with ID: C1
Number of Objects in Cluster 1	1809

Data Objects in the Cluster 1

b) Cluster 2

Cluster objects, those are in the ε-neighborhood of any one of the Core Objects at all the discovered LEVELS. These objects themselves may or may not be CORE. They are the 'Density Connected' objects with the core object having **Customer ID = C12**.

Cluster 2	
Cluster Representative	Data Object (Customer) with ID: C12
Number of Objects in Cluster 1	647

Data Objects in the Cluster 2

c) Cluster 3

Cluster objects, those are in the ε-neighborhood of any one of the Core Objects at all the discovered LEVELS. These objects themselves may or may not be CORE. They are the 'Density Connected' objects with the core object having **Customer ID = C16**.

Cluster 3	
Cluster Representative	Data Object (Customer) with ID: C16
Number of Objects in Cluster 1	487

Data Objects in the Cluster 3

d) Cluster 4

Cluster objects, those are in the ε-neighborhood of any one of the Core Objects at all the discovered LEVELS. These objects themselves may or may not be CORE. They are the 'Density Connected' objects with the core object having **Customer ID = C13**.

Cluster 4	
Cluster Representative	Data Object (Customer) with ID: C13
Number of Objects in Cluster 1	389

Data Objects in the Cluster 4

e) Cluster 5

Cluster objects, those are in the ε-neighborhood of any one of the Core Objects at all the discovered LEVELS. These objects themselves may or may not be CORE. They are the 'Density Connected' objects with the core object having **Customer ID = C3**.

Cluster 5	
Cluster Representative	Data Object (Customer) with ID: C3
Number of Objects in Cluster 1	744

Data Objects in the Cluster 5

f) Cluster 6

Cluster objects, those are in the ε-neighborhood of any one of the Core Objects at all the discovered LEVELS. These objects themselves may or may not be CORE. They are the 'Density Connected' objects with the core object having **Customer ID = C20**.

Cluster 6	
Cluster Representative	Data Object (Customer) with ID: C20
Number of Objects in Cluster 1	374

Data Objects in the Cluster 6

g) Cluster 7

Cluster objects, those are in the ε-neighborhood of any one of the Core Objects at all the discovered LEVELS. These objects themselves may or may not be CORE. They are the 'Density Connected' objects with the core object having **Customer ID = C54**.

Cluster 7	
Cluster Representative	Data Object (Customer) with ID: C54
Number of Objects in Cluster 1	306

Data Objects in the Cluster 7

h) Outliers

When the clustering processing ceases and no further group of the data objects can full fill the clustering parameters; then the remaining data objects are termed as the Outliers. These objects could not become part of any one of the discovered clusters till completion of the process. In our dataset of 5000 objects, there are **244** Outlying objects.

Summary of the Clustering Process

On successful completion of the clustering process following clusters were discovered:

Table 4.5: Summary of the Clustering Process

Cluster No.	Cluster Representative (Core Object)	Cluster objects
Cluster 1	Data Object (Customer) with ID: C01	1809
Cluster 2	Data Object (Customer) with ID: C12	647
Cluster 3	Data Object (Customer) with ID: C16	487
Cluster 4	Data Object (Customer) with ID: C13	389
Cluster 5	Data Object (Customer) with ID: C03	744
Cluster 6	Data Object (Customer) with ID: C20	374
Cluster 7	Data Object (Customer) with ID: C54	306
	Total Objects In all the Clusters	**4756**
Outliers	5000 - 4756	**244**

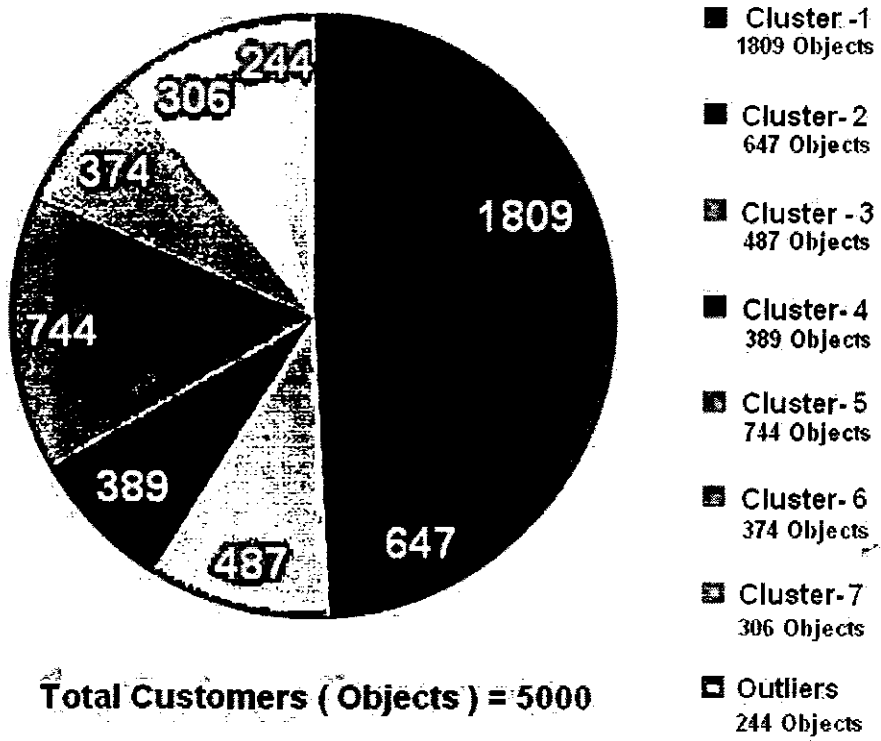


Figure 4.5: Pie - Chart of the Clustering Results

Chapter 5

Conclusion & Future Work

5.1 Conclusions

- This research work was initiated with an aim to enhance one of the in use data clustering algorithms, in terms of its sensitivity to the privacy of an individual's record. In others words it can be said that it was an attempt of 'value addition' to one of the existing algorithm. I selected one of the Density Based algorithms – the DBSCAN (Density Based Spatial Clustering of Applications with Noise), and employed a combination of the data encryption techniques to preserve privacy of the data while applying the DBSCAN. The result shown in the last section is an evidence of achievement of the goal.

- The value added or privacy preserving DBSCAN which has been implemented in this research work, can be applied to the dataset requiring a certain level of data privacy along with the Clustering process.

- The employed data encryption techniques can be tailored according to required level of data privacy. In case of a highly sensitive domain it is recommended to enhance the complexity of the linear data transformation, but it is to be kept in mind that more complex data transformation increases the loss of computational value of the data records, which ultimately disturbs accuracy of the output. In fact the process data hiding or data encryption and the accuracy of the mining results are inversely proportional. Increase on one side decrease the other side.

- Output or the resultant clusters in the Density Based Clustering change significantly due to a change the clustering parameters (e-neighborhood and Minimum Points). Selection of these attributes requires a meticulous analysis of the dissimilarity matrix. Only a wise selection of the attributes would result into the truly representative clusters after the data clustering process. While implementing the algorithm I had the sample clustering with five different values of the e-neighborhood. It was observed that the variance of the output clusters depends on the statistical properties of the under processing data like the variance of the computed loyalty or profitability of an individual customer.

- The Minimum Points (MinPoints) also play crucial role in the density based clustering process. The clustering process may halt without covering the complete dataset with a single fixed value of MinPoints. In such case the halted clustering process can be pushed forward by reducing the MinPoints (while keeping the e-neighborhood constant) for the data items those are yet to be clustered.

- Computation of the *Data Dissimilarity Matrix* (*DDM*) has been proved a very resource hungry job as it requires processors with ultra high speed and large amount of main memory. Luckily it is possible to compute the DDM in parallel to divide the computational load on multiple machines. Its parallel computation is highly recommended.

- Data Miners, experts and scholars of this field are familiar with the well known problem of the Data Clustering-- "The Curse of Dimensionalities". The severity of this problem increases with increase in size of the under processing data. Most of the Data Management Systems are unable to handle the large number of dimensions (attributes or columns). For example the famous DBMS --- ORACLE (version 9i, Release 9.2.0.2.0) allows only 1000 dimension for a single relation while even a dataset of 5000 items requires a dissimilarity matrix with size 5000 x 5000 !, this problem yet to be addressed.

- Selection of the correct, complete and free of noise an essential prerequisite to get the true clustering results. Even a small amount of noise in an individual record may lead the clustering algorithm to include the record in an inappropriate cluster.

- In order to preserve data privacy, it is highly recommended that *Attribute Filtering / Attribute Hiding* should be preferred to the linear transformation of data. Especially in case of the *neutral attributes*-- those are not used or of minor significance in calculation of the dissimilarity matrix before starting the clustering process.

5.2 Future work

- As it is said earlier that the change in the clustering parameters (e-neighborhood and Minimum Points) significantly change the clustering results. On the other hands the available text does not present any formal method compute the parameters those may be declared as the 'true' parameters, such variance of result cannot be tolerated. The density based algorithm requires improvement and some guidelines for the miners to get the resultant clusters with acceptable difference.

- Selection of a particular Distance Measure to compute the Data Dissimilarity Matrix is majorly depends on the data miner. Being in a constrained environment (having casual data processing machine with just 1 gigabyte main memory) I preferred it to use the Manhattan Distance (or City Block Distance) as it requires less time and memory to compute the DDM. The DDM may be computed with the other well known distance measures like (Euclidian Distance Measure or Minkowski Distance Measure) to compare the variation in the resultant clusters.

- Visualization of the output clusters is usually listed among "issues of the data mining", same was the case with this clustering experiment. I was not in a position to develop some application to visualize the resultant clusters due to lack of the programming skills and shortage of time. Although the visualization of the clustering results facilitates the end users or the non technical user to comprehend the results.

References

References

[1] Carolyn Begg, "Database Systems" by Thomas Connolly, 3rd Edition, PEARSON Education.

[2] David Hand, Heikki Mannila and Padhraic Smyth, "Principles of Data Mining" ISBN: 026208290x, the MIT Press. 2001.

[3] Y. Lindell and B. Pinkas, Privacy Preserving Data Mining, Advances in Cryptology - CRYPTO '00. Lecture Notes in Computer Science, Springer-Verlag Vol. 1880, pages. 36–53, 2000.

[4] S. Oliveira and O. Zaiane. Privacy Preserving Clustering By Data Transformation. In Proc. of the 18th Brazilian Symposium on Databases, pages 304-318, Manaus, Brazil, October 2003.

[5] Sushil Jajodia, et. al. "The Inference Problem", Center for Secure Information Systems George Mason University, 2003.

[6] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu. "A density-based algorithm for discovering clusters in large spatial databases with noise". In Evangelos Simoudis, Jiawei Han, Usama M. Fayyad. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), 1996.

[7] Jiawei Han and Micheline Kamber, "Data Mining: Concepts and Techniques Morgan", 2nd Edition. Kaufmann Publishers, San Francisco, CA. 2006.

[8] Csilla Farkas et. al, "The Inference Problem: A Survey", Department. of Computer Science and Engineering, University of South Carolina. SIGKDD Explorations, 2003.

[9] Nan Zhang, "Towards Comprehensive Privacy Protection in Data", Department of Computer Science and Engineering, University of Texas at Arlington, Arlington, TX 76019-0015, USA, 2007.

[10] Z. Huang and W. Du and B. Chen, "Deriving Private Information from Randomized Data," in Proceedings of the 2005 ACM SIGMOD Conference, pp. 37-48, Baltimroe, MD, 2005.

[11] Jun-Lin Lin, "Privacy Preserving Clustering by K-Anonymity technique" Department of Information Management, Yuan Ze University Chung-Li, Taiwan. PAIS'08, Nantes France 2008.

[12] C. Chiu and C.-Y. Tsai, "A k-anonymity clustering method for effective data privacy preservation". In Third International Conference on Advanced Data Mining and Applications (ADMA), 2007.

[13] Ashwin Machanavajjhala, Daniel Kifer, et al. "L-Diversity: Privacy Beyond k-Anonymity"; Cornell University. ACM Trans. Knowl. Discov. Data 1, 1, Article 3, March 2007.

[14] Jun-Lin Lin, Yuan Ze; "Privacy Preserving Item-set Mining through Fake Transactions", SAC'07 March 1115, Seoul, Korea, 2007.

[15] Xiaodong Lin, Chris Clifton and Michael Zhu. "Privacy-preserving clustering with distributed EM mixture modeling". Springer-Verlag London Ltd. 2004, Knowledge and Information Systems, 2005.

[16] Mahir Can Doganay et al. "Distributed Privacy Preserving k-Means Clustering with Additive Secret Sharing". PAIS'08, Nantes, France, 2008.

[17] Dr.A.M.Natarajan R.R.Rajalaxmi, "A Hybrid Data Transformation Approach for Privacy Preserving Clustering". T. Sobh(ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, Springer, pages 403–408, 2007.

[18] Osmar R. Zaiane etl. "A Privacy-Preserving Clustering by Object Similarity-Based Representation and Dimensionality Reduction Transformation", Department of Computing University of Alberta Edmonton, AB, Canada, 2007.

[19] Srujana Merugu and Joydeep Ghosh. "Privacy-preserving Distributed Clustering using Generative Models". Electrical and Computer Engineering University of Texas, Austin, 2005.

[20] Srujana Merugu and Joydeep Ghosh. "Privacy Preserving Unsupervised Clustering over Vertically Partitioned Data". SIGKDD '03, Washington, DC, USA, 2003.

[21] Jun-Lin, "Privacy Preserving Clustering by K-Anonymity technique", Department of Information Management, Yuan Ze University Chung-Li, Taiwan. PAIS'08, Nantes, France, March 29, 2008.

[22] Stanley R. M. Oliveira, Embrapa Information Technology, Brasil. Osmar R. Zaiane, "Privacy Preserving Frequent Itemset Mining" University of Alberta, Edmonton, Canada, 2002.

[23] R. Agrawal and R. Srikant, "Privacy Preserving Mining of Association Rules". In Proceedings of the SIGKDD Edmonton, Alberta, Canada, 2002.

[24] Jiawei Han and Micheline Kamber, "Data Mining: Concepts and Techniques Morgan", 2nd Edition. Kaufmann Publishers, San Francisco, CA,2006.