## IMPLEMENTATION OF K-MEANS AND K-MEDOIDS CLUSTERING ALGORITHMS

## B.Tech. Final Year Project Report

## BY

**ANISHA PAL**



## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

## NEOTIA INSTITUTE OF TECHNOLOGY, MANAGEMENT & SCIENCE

## JHINGA, AMIRA, D.H.ROAD, 24-PARAGANAS (S) - 743368, WB (INDIA)

## May, 2017

## Implementation of K-means and K-medoids clustering algorithms and Elbow method to determine the optimal k

## A Major Project Report

### Submitted in partial fulfillment of the requirements for the award of the degree

*Of*

#### Bachelor of Technology

*In*

##### **COMPUTER SCIENCE AND ENGINEERING**

## BY

|  |  |  |
| --- | --- | --- |
| **ANISHA PAL** | **14400113005** | **131440110005** |

**Under the Guidance of**

Prof. Subrata Bose

## imgpsh_fullsize

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

## NEOTIA INSTITUTE OF TECHNOLOGY, MANAGEMENT & SCIENCE

## JHINGA, AMIRA, D.H.ROAD, 24-PARAGANAS (S) - 743368, WB (INDIA)

## May, 2017

**Project Report Approval**

The project report entitled

## Implementation of K-means and K-medoids clustering algorithms and Elbow method to determine the optimal ‘k’

Submitted by

**Anisha Pal**

Roll No: 14400113005

Reg. No: 131440110005

Is approved for the degree of

**Bachelor of Technology**

In

**Computer Science and Engineering**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| Prof. Subrata Bose |

***Head of the Department***

Department of Computer Science & Engineering

NITMAS

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Head of the Department***

Department of Computer Science & Engineering

NITMAS



# imgpsh_fullsize

# CERTIFICATE OF ORIGINALITY

I hereby certify that the work which is being presented in the B.Tech. Final Year Project Report entitled **“IMPLEMENTATION OF K-MEANS AND K-MEDOIDS CLUSTERING ALGORITHMS AND ELBOW METHOD TO DETERMINE THE OPTIMAL ‘K’ ”,** in partial fulfilment of the requirements for the award of the **Bachelor of Technology in Computer Science & Engineering** and submitted to the Department of Computer Science & Engineering of Neotia Institute of Technology, Management & Science, West Bengal is an authentic record of my own work carried out from July, 2016 to November, 2016 under the supervision of **Prof. Subrata Bose**.

The matter presented in this thesis has not been submitted by me for the award of any other degree elsewhere.

|  |
| --- |
| **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  *Signature of Candidate* |
| **Anisha Pal** |
| **14400113005** |
| **131440110005** |



**CERTIFICATE OF RECOMMENDATION**

This is to certify that the Project entitled “IMPLEMENTATION OF K-MEANS AND K-MEDOIDS CLUSTERING ALGORITHMS AND ELBOW METHOD TO DETERMINE THE OPTIMAL ‘K’ ” has been submitted by **MS.** **ANISHA PAL** under my guidance in partial fulfilment of the degree of Bachelor of Technology in Computer Science & Engineering of Neotia Institute of Technology, Management & Science, Jhinga, WB during the academic year 2016-2017.

|  |
| --- |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| *Signature of Supervisor* |
| Prof. Subrata Bose |

***Head of the Department***

Department of Computer Science & Engineering

NITMAS

Date:

Place:



**ACKNOWLEDGEMENT**

This Project has been carried out to meet the academic requirements of Maulana Abul Kalam Azad University of Technology, West Bengal. I would like to put on record, my appreciation and gratitude to all who have rendered their support and guidance. Without them, it would not have been possible for me to shape this project.

I have received immense guidance from my project supervisor Mr. Subrata Bose, Professor and Head of the Computer Science Engineering Department, NITMAS. I would therefore like to convey my sincere gratitude to him. Finally I want to pay my respect to my parents for their love, affection and motivation during my project work.

Anisha Pal

25th of May, 2017

**INDEX**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl No** | | **Topic** | **Page** |
| 1 | Abstract ……………………………………………………………………………......... | | 9 |
| 2 | Introduction ……………………………………………………………………………... | | 10 |
|  | 2.1 What is data mining? ………………………………………………………………  2.2 Why Data Mining is important……………………………………………………... | |  |
| 3 | Cluster Analysis and Classification of Clustering Techniques……………………... | | 12 |
|  | 3.1 | Application examples of Cluster Analysis ……..…………………………….... | 13 |
|  | 3.2 | Different types of Algorithms for clustering …………………………………… | 14 |
| 4 | Partitioning Clustering: Basic Concept and its Classification ……………………… | | 16 |
|  | 4.1 Centroid Based Technique: The K-means clustering Method ………………... | | 17 |
|  | 4.2 Object Based Technique: The K-medoids clustering Method ………………… | | 21 |
| 5 | Finding the optimal K in clustering …….……………………………………….......... | | 25 |
|  | 5.1 | Elbow Method: Using Sum of Squared Error Method ………………………. | 25 |
|  | 5.2 | Elbow Method: Obtaining the elbow point in the project ……………………. | 26 |
|  |  | 5.3.1 SSE Value generation and ELBOW graph: K-means ………………… | 27 |
|  |  | 5.3.2 SSE Value generation and ELBOW graph: K-medoids ………………. | 31 |
| 6 | Conclusion ……………………………………………………………………………… | | 37 |
| 7 | Limitations ……………………………………………………………………............... | | 38 |
| 8 | References ……………………………………………………………………………… | | 39 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Fig** | **Topic** | **Page** |
| 1 | Partitioning Algorithm ………………………………………………………………… | 13 |
| 2 | Hierarchical based Algorithm …………………………….………………………….. | 14 |
| 3 | Density based Algorithm ……………………………………………………………... | 14 |
| 4 | Grid based Algorithm ……………………………………..………………………….. | 15 |
| 5 | Elbow method graph ……………………………………...………………………….. | 20 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Fig** | **Topic** | **Page** |
| 1 | Partitioning Algorithm ………………………………………………………………… | 13 |
| 2 | Hierarchical based Algorithm …………………………….………………………….. | 14 |
| 3 | Density based Algorithm ……………………………………………………………... | 14 |
| 4 | Grid based Algorithm ……………………………………..………………………….. | 15 |
| 5 | Steps of K-means Clustering Algorithm ………………...………………………….. | 18 |
| 6 | Elbow method graph ……………………………………...………………………….. | 20 |

**ABSTRACT**

Data clustering is an unsupervised data analysis and data mining technique, which offers reﬁned and more abstract views to the inherent structure of a data set by partitioning it into a number of disjoint or overlapping (fuzzy) clusters. Hundreds of clustering algorithms have been developed by researchers from a number of different scientiﬁc disciplines. The intention of this report is to present a special class of clustering algorithms, namely K-means and K-medoids (under partitioning based), where cluster validation is an important step and it is evaluated based on the similarity between two clusters. Cluster analysis is used everywhere be it during public census, the classification of species of animals and plants or similar diagnostic cases.

**CHAPTER 1**

**INTRODUCTION**

This work is the study of k-means and k-medoids clustering algorithms. Clustering is one of data mining principles. Therefore, we begin with the introduction of data mining and then get into clustering.

**2.1 What is data mining?**

Simply stated, data mining refers to extracting or “mining” knowledge from large amounts of data. The term is actually a misnomer. Remember that the mining of gold from rocks or sand is referred to as gold mining rather than rock or sand mining. Thus, “data mining” should have been more appropriately named “knowledge mining from data”, which is unfortunately somewhat long. “Knowledge mining”, a shorter term, may not reflect the emphasis on mining from large amounts of data. Nevertheless, mining is a vivid term characterizing the process that finds a small set of precious nuggets from a great deal of raw material. Thus, such a misnomer which carries both “data” and “mining” became a popular choice. There are many other terms carrying a similar or slightly different meaning to data mining, such as knowledge mining from databases, knowledge extraction, data/pattern analysis, data archaeology, and data dredging.

Data mining (sometimes called data or knowledge discovery) is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cut costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases

Although data mining is a relatively new term, the technology is not. Companies have used powerful computers to sift through volumes of supermarket scanner data and analyze market research reports for years. However, continuous innovations in computer processing power, disk storage, and statistical software are dramatically increasing the accuracy of analysis while driving down the cost.

Many people treat data mining as a synonym for another popularly used term, Knowledge Discovery from Data, or KDD. Alternatively, others view data mining as simply an essential step in the process of knowledge discovery.

Knowledge discovery consists of an iterative sequence of the following steps:

1. **Data cleaning** (to remove noise and inconsistent data)
2. **Data integration** (where multiple data sources may be combined)
3. **Data selection** (where data relevant to the analysis task are retrieved from the database)
4. **Data transformation** (where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations, for instance)
5. **Data mining** (an essential process where intelligent methods are applied in order to extract data patterns)
6. **Pattern evaluation** (to identify the truly interesting patterns representing knowledge based on some

interestingness measures)

1. **Knowledge presentation** (where visualization and knowledge representation techniques are used to present the mined knowledge to the user)

**2.2 Why Data Mining is important?**

The major reason that data mining has attracted a great deal of attention in information industry in recent years is due to the wide availability of huge amounts of data and the imminent need for turning such data into useful information and knowledge. The information and knowledge gained can be used for applications ranging from business management, production control, and market analysis, to engineering design and science exploration.

Data mining can be viewed as a result of the natural evolution of information technology. An evolutionary path has been witnessed in the database industry in the development of the functionalities such as data collection and database creation, data management (including data storage and retrieval, and database transaction processing), and data analysis and understanding (involving data warehousing and data mining).

For instance, the early development of data collection and database creation mechanisms served as a prerequisite for later development of effective mechanisms for data storage and retrieval, and query and transaction processing. With numerous database systems offering query and transaction processing as common practice, data analysis and understanding has naturally become the next target. Since the 1960's, database and information technology has been evolving systematically from primitive file processing systems to sophisticated and powerful databases systems.

The research and development in database systems since the 1970's has led to the development of relational database systems (where data are stored in relational table structures), data modeling tools, and indexing and data organization techniques. In addition, users gained convenient and flexible data access through query languages, query processing, and user interfaces. Efficient methods for on-line transaction processing (OLTP), where a query is viewed as a read-only transaction, have contributed substantially to the evolution and wide acceptance of relational technology as a major tool for efficient storage, retrieval, and management of large amounts of data.

#### Data

#### Data are any facts, numbers, or text that can be processed by a computer. Today, organizations are accumulating vast and growing amounts of data in different formats and different databases. This includes:

* Operational or transactional data such as, sales, cost, inventory, payroll, and accounting
* Non-operational data, such as industry sales, forecast data, and macro-economic data
* Metadata - data about the data itself, such as logical database design or data dictionary definitions

#### Information

The patterns, associations, or relationships among all this data can provide information.

#### Knowledge

Information can be converted into knowledge about historical patterns and future trends. For example, summary information on retail supermarket sales can be analyzed in light of promotional efforts to provide knowledge of consumer buying behavior. Thus, a manufacturer or retailer could determine which items are most susceptible to promotional efforts.

#### Data Warehouses

Dramatic advances in data capture, processing power, data transmission, and storage capabilities are enabling organizations to integrate their various databases into data warehouses. Data warehousing is defined as a process of centralized data management and retrieval. Data warehousing, like data mining, is a relatively new term although the concept itself has been around for years. Data warehousing represents an ideal vision of maintaining a central repository of all organizational data. Centralization of data is needed to maximize user access and analysis. Dramatic technological advances are making this vision a reality for many companies. And, equally dramatic advances in data analysis software are allowing users to access this data freely. The data analysis software is what supports data mining.

**CHAPTER 3**

**CLUSTER ANALYSIS AND CLASSIFICATION OF CLUSTERING TECHNIQUES**

Cluster analysis (or clustering, datasegmentation) is finding the similarities between data according to the characteristics found in the data and grouping similar data objects into clusters

It is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense or another) to each other than to those in other groups (clusters). It is a main task of exploratory [data mining](https://en.wikipedia.org/wiki/Data_mining), and a common technique for [statistical](https://en.wikipedia.org/wiki/Statistics) [data analysis](https://en.wikipedia.org/wiki/Data_analysis), used in many fields, including [machine learning](https://en.wikipedia.org/wiki/Machine_learning), [pattern recognition](https://en.wikipedia.org/wiki/Pattern_recognition), [image analysis](https://en.wikipedia.org/wiki/Image_analysis), [information retrieval](https://en.wikipedia.org/wiki/Information_retrieval), [bioinformatics](https://en.wikipedia.org/wiki/Bioinformatics), [data compression](https://en.wikipedia.org/wiki/Data_compression), and [computer graphics](https://en.wikipedia.org/wiki/Computer_graphics).

**3.1 Application examples of Cluster Analysis**

* Clustering analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing.
* Clustering can also help marketers discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns.
* In the field of biology, it can be used to derive plant and animal taxonomies, categorize genes with similar functionalities and gain insight into structures inherent to populations.
* Clustering also helps in identification of areas of similar land use in an earth observation database. It also helps in the identification of groups of houses in a city according to house type, value, and geographic location.
* Clustering also helps in classifying documents on the web for information discovery.
* Earth-quake studies: Observed earth quake epicenters should be clustered along continent faults
* Climate: understanding earth’s climate, finding patterns of atmospheric and ocean
* Economic Science: market research.
* Data reduction
  + Summarization: Preprocessing for regression, PCA, classification, and association analysis
  + Compression: Image processing: vector quantization
* Hypothesis generation and testing
* Prediction based on groups
  + Cluster & find characteristics/patterns for each group
* Finding K-nearest Neighbors
  + Localizing search to one or a small number of clusters
* Outlier detection: Outliers are often viewed as those “far away” from any cluster

**3.2 Different Types of Algorithms or Approaches for clustering**

There are different methods of clustering, known as:

* **Partitioning approach**:
  + Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square errors
  + Typical methods: k-means, k-medoids, CLARA, CLARAN

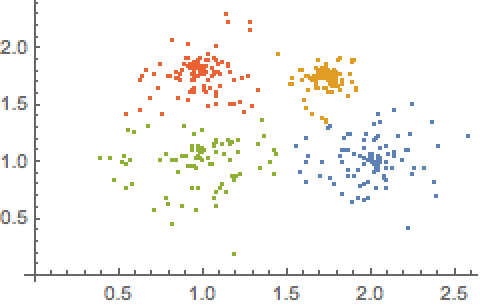


Fig. 1 Partitioning Algorithm

* **Hierarchical approach:** 
  + Create a hierarchical decomposition of the set of data (or objects) using some criterion
  + Typical methods: Diana, Agnes, BIRCH, CAMELEON

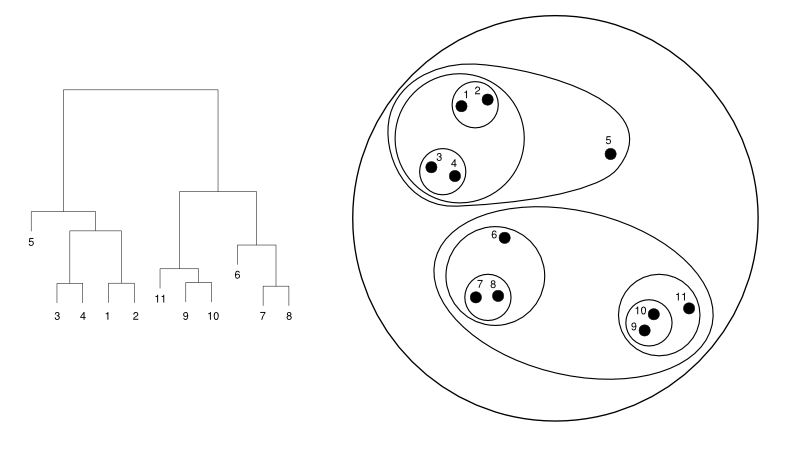


Fig.2 Hierarchical based Algorithm

* **Density-based approach:** 
  + Based on connectivity and density functions
  + Typical methods: DBSACN, OPTICS, DenClue

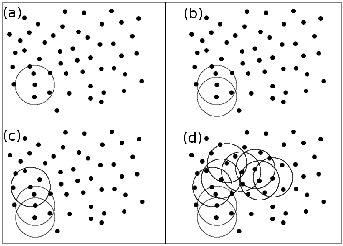


Fig.3 Density-based Algorithm

* **Grid-based approach**:
  + Based on a multiple-level granularity structure
  + Typical methods: STING, WaveCluster, CLIQUE

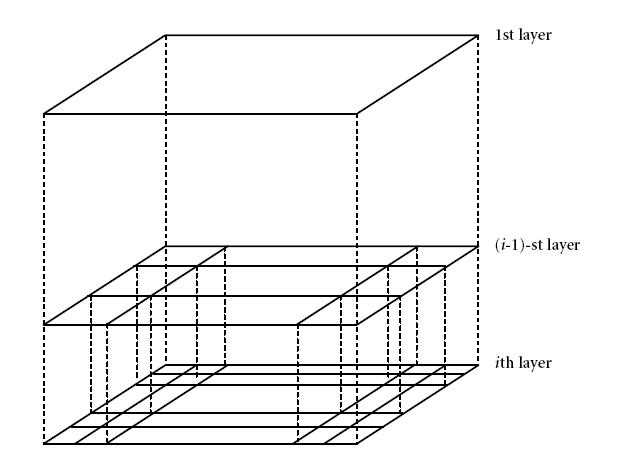


Fig.4 Grid-based Algorithm

**CHAPTER 4**

**PARTITIONING CLUSTERING: BASIC CONCEPT**

**AND IT’S CLASSIFICATIONS.**

Suppose we are given a database‘d’ of ‘n’ objects and the partitioning method constructs ‘k’ partition of data(clusters). Each partition will represent a cluster and k ≤ n. It means that it will classify the data into k groups, which satisfy the following requirements −

* Each group contains at least one object.
* Each object must belong to exactly one group.
* The sum of squared distances is minimized (where ci is the centroid or medoid of cluster Ci)



**Points to remember**

* For a given number of partitions (say k), the partitioning method will create an initial partitioning.
* Then it uses the iterative relocation technique to improve the partitioning by moving objects from

one group to other.

* Given *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion
  + Global optimal: exhaustively enumerate all partitions
  + Heuristic methods: *k-means* and *k-medoids* algorithms
  + *K-means* (Centroid based) (MacQueen’67, Lloyd’57/’82): Each cluster is represented by the center of the cluster
  + *K-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw’87): Each cluster is represented by one of the objects in the cluster.

**4.1 CENTROID BASED TECHNIQUE: THE**

***K-MEANS* CLUSTERING METHOD**

**K-means clustering** is a method of [vector quantization](https://en.wikipedia.org/wiki/Vector_quantization), originally from [signal processing](https://en.wikipedia.org/wiki/Signal_processing), that is popular for [cluster analysis](https://en.wikipedia.org/wiki/Cluster_analysis) in [data mining](https://en.wikipedia.org/wiki/Data_mining). *K-means* clustering aims to [partition](https://en.wikipedia.org/wiki/Partition_of_a_set) *n* observations into *k* clusters in which each observation belongs to the [cluster](https://en.wikipedia.org/wiki/Cluster_(statistics)) with the nearest [mean](https://en.wikipedia.org/wiki/Mean), serving as a [prototype](https://en.wikipedia.org/wiki/Prototype) of the cluster.

K-means is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori.

**4.1.1 How does the K-means method works?**

The k-means algorithm proceeds as follows:

1. It randomly selects k of the objects, each of which initially represents a cluster mean or center.

2. For each of the remaining objects, an object is assigned to the cluster to which it is the most similar

based on the distance between the object and the cluster mean.

3. It then computes the new mean for each cluster.

4. This process iterates until the criterion function converges.

Typically, the square-error criterion is used, defined as

Where ‘E’ is the sum of the square error for all objects in the data set

‘p’ is the point in space representing a given object

‘mi’ is the mean of cluster Ci (both p and mi are multidimensional).

In other words, for each object in each cluster, the distance from the object to its cluster center is squared, and the distances are summed. This criterion tries to make the resulting k clusters as compact and as separate as possible.

**4.1.2 Algorithm: K-means.**

The k-means algorithm for partitioning, where each cluster’s center is represented by the mean value of the objects in the cluster.

**Input**: k: the number of clusters, D: a data set containing n objects.

**Output**: A set of k clusters.

**Method**:

* + - 1. Arbitrarily choose k objects from D as the initial cluster centers.
      2. Repeat.
      3. (Re) assign each object to the cluster to which the object is the most similar,

based on the mean value of the objects in the cluster.

* + - 1. Update the cluster means, i.e., calculate the mean value of the objects for each cluster.
      2. Until no change.

**4.1.3 EXPERIMENTAL RESULT**

In the project, I have taken 12 inputs of a 2- Dimension information (height, weight).

**Input:**

Elements entered as input**:**

|  |  |
| --- | --- |
| 185 | 72 |
| 170 | 56 |
| 168 | 60 |
| 179 | 68 |
| 182 | 72 |
| 188 | 77 |
| 180 | 71 |
| 180 | 70 |
| 183 | 84 |
| 180 | 88 |
| 180 | 67 |
| 177 | 76 |

No. of clusters to be taken: 4

**Objective Code:**

1. During the first pass ,the first k no. of elements from the input elements are taken (where k is the cluster no. that is taken through input) and each of these selected elements are considered to be the respective mean or centroid of each corresponding clusters.

// initializing the first means

for(i=0;i<k;i++)

{

mean[i][0] = pts[i][0];

mean[i][1] = pts[i][1]; // where pts[][] is any element of two co ordinates

}

2. The distance between the centroid and the elements are calculated and the clusters are formed:

for(i=0;i<n;i++)

{

min = 5000000.00;

for(j=0;j<k;j++)

{

// finding the element nearest to mean by Euclidean distance

// formula: sqrt((a1 - b1)^2 + (a2 - b2)^2)

d = sqrt(pow((mean[j][0] - pts[i][0]),2) + pow((mean[j][1] - pts[i][1]),2));

if(d < min)

{

min = d;

min\_index = j;

}

}

top[min\_index]++;

c[min\_index][top[min\_index]] = i;

}

3. After the first pass, the mean of a particular cluster is calculated

**// mean calculation**

**for(i=0;i<k;i++)**

**{**

**if (top[i]==-1)**

**break;**

**float sum\_x = 0;**

**float sum\_y = 0;**

**for(j=0;j<=top[i];j++)**

**{**

**sum\_x += pts[c[i][j]][0];**

**sum\_y += pts[c[i][j]][1];**

**}**

**mean[i][0] = sum\_x/j;**

**mean[i][1] = sum\_y/j;**

**}**

4. This procedure goes on while () loop and stops when the cluster elements before the previous pass and the existing pass becomes equal.

**Output:**

CLUS# MEAN CLUSTER

-------- ---------- ------------

1 183.67, 83.00 {(188.00, 77.00), (183.00, 84.00), (180.00, 88.00)}

2 170.00, 56.00 {(170.00, 56.00)}

3 168.00, 60.00 {(168.00, 60.00)}

4 180.43, 70.86 {(185.00, 72.00), (179.00, 68.00), (182.00, 72.00), (180.00, 71.00),

(180.00, 70.00), (180.00, 67.00), (177.00, 76.00)}

**4.1.4 Problem of the K-Means Method**

* **Sensitive to scale**

Rescaling your datasets will completely change the results. While this it is not bad, not realizing that you have to spend extra attention to scaling your data is bad.

* **Even on perfect data sets, it can get stuck in a local minimum**
* **Means are continuous**
* **Too easy to use badly**

All in all, it's too easy to throw k-means on the data, and nevertheless get a result out (that is pretty much random, but it is not noticeable). It would be better to have a method which can fail if you haven't understood the data.

**4.2****OBJECT BASED TECHNIQUE: THE**

***K-MEDOIDS* CLUSTERING METHOD**

The k-means algorithm is sensitive to outliers. Since an object with an extremely large value will substantially distort the distribution of the data

Instead of taking the mean value of the object in a cluster as a reference point, medoids can be used, which is the most centrally located object in a cluster

The **k-medoids algorithm** is a [clustering](https://en.wikipedia.org/wiki/Data_clustering) [algorithm](https://en.wikipedia.org/wiki/Algorithm) related to the [k-means](https://en.wikipedia.org/wiki/K-means) algorithm and the medoid shift algorithm.

Both the k-means and k-medoids algorithms are partitioned (breaking the dataset up into groups) and both attempt to minimize the distance between points labelled to be in a cluster and a point

designated as the center of that cluster.

In contrast to the k-means algorithm, k-medoids chooses data points as centers ([medoids](https://en.wikipedia.org/wiki/Medoids) or exemplars) and works with an arbitrary metric of distances between data points.

**4.2.1 How might the algorithm be modified to diminish such sensitivity?**

Instead of taking the mean value of the objects in a cluster as a reference point, we can pick actual objects to represent the clusters, using one representative object per cluster. Each remaining object is clustered with the representative object to which it is the most similar. The partitioning method is then performed based on the principle of minimizing the sum of the dissimilarities between each object and its corresponding reference point. That is, an absolute-error criterion is used, defined as

Where E is the sum of the absolute error for all objects in the data set;

p is the point in space representing a given object in cluster Cj ;

and oj is the representative object of Cj .

In general, the algorithm iterates until, eventually, each representative object is actually the medoid, or most centrally located object, of its cluster. This is the basis of the k-medoids method for grouping n objects into k clusters

**4.2.2 Algorithm: K-medoids**

A k-medoids algorithm for partitioning based on medoid or central objects.

**Input:** k: the number of clusters, D: a data set containing n objects.

**Output**: A set of k clusters.

**Method**:

1. arbitrarily choose k objects in D as the initial representative objects or seeds
2. repeat
3. assign each remaining object to the cluster with the nearest representative object
4. randomly select a non-representative object, orandom
5. compute the total cost, S, of swapping representative object, oj , with orandom
6. if S < 0 then swap oj with orandom to form the new set of k representative objects
7. until no change

**4.2.3 EXPERIMENTAL RESULT**

In the project, I have taken 12 inputs of a 2- Dimension information (height, weight).

**Input:**

Same elements taken at early experiment has been entered

No. of clusters to be taken: 4

**Objective Code:**

1. At first, random choosing of K-Medoids and assigning them to nth medoid.

for(int i=0;i<k;i++)

{

int temp = rand()%n;

medoids2.push\_back(temp);

}

1. Calculating the cost of current configuration

int temp = medoids2[i]; bool flag = false;

float c1 = calculateCost(medoids2, n, k);

float calculateCost(vector<int> &medoid, int n, int k)

{

float cost = 0;

for(int i=0;i<n;i++)

{

float min = LONG\_LONG\_MAX;

for(int j=0;j<k;j++)

{

float d = fabs(pts[medoid[j]][0] - pts[i][0]) + fabs(pts[medoid[j]][1] - pts[i][1]);

if(d < min)

min = d;

}

cost += min;

}

return cost;

}

3. Calculating the cost of configuration

temp\_medoid = medoids2;

for(j=0;j<n;j++)

{

if(temp == j)

continue;

temp\_medoid[i] = j;

float c2 = calculateCost(temp\_medoid, n, k);

4. Re-computing till the best configuration

if(c2<c1)

{

medoids2[i] = j;

flag = true;

break;

}

}

5. The distance between the selected object and the elements are calculated and the clusters are formed:

for(j=0;j<k;j++)

{

float d = fabs(pts[medoids2[j]][0] - pts[i][0]) + fabs(pts[medoids2[j]][1] - pts[i][1]);

if(d < min)

{

min = d;

cluster\_number = j;

}

}

top[cluster\_number] += 1;

cluster[cluster\_number][top[cluster\_number]] = i; }

6. The procedure goes on while () till there's no change in medoid configuration.

**Output:**

CLUS# OBJECT CLUSTER

-------- ---------- ------------

1 170.0, 56.00 {(170.00, 56.00), (168.00, 60.00)}

2 183.00, 84.00 {(183.00, 84.00), (180.00, 88.00)}

3 180.00, 70.00 {(179.00, 68.00), (180.00, 71.00), (180.00, 70.00), (180.00,

67.00), (177.00, 76.00)}

4 185.00, 72.00 {(185.00, 72.00), (188.00, 77.00)}

**CHAPTER 5**

**FINDING THE OPTIMAL K IN CLUSTERING**

Partition, especially K-Means clustering, attracts a most considerable interest in the literature, and the issue of properly defining or determining number of clusters K is attacked by dozens of researchers. Most of them accept the square error criterion that is alternatingly minimized. The estimation of the optimal number of clusters within a set of data points is a very important problem, as most clustering algorithms need that parameter as input in order to group the data.

There are many approaches to choosing the number of clusters in a partition.

In this project, we have emphasized on the **Elbow method** which is a variance based approach using intuitive or model based functions. This method exists upon the idea that one should choose a number of clusters so that adding another cluster doesn't give much better modelling of the data

**5.1 ELBOW method: Using Sum of Squared Error function**

The idea of the elbow method is to run k-means or k-medoids clustering on the dataset for a range of values of k (say, k from 1 to 10 in the examples above), and for each value of k calculate the sum of squared errors (SSE). Then, plotting a line chart of the SSE for each value of k. If the line chart looks like an arm, then the "elbow" on the arm is the value of k that is the best.

The idea is that we want a small SSE, but that the SSE tends to decrease toward 0 as we increase k (the SSE is 0 when k is equal to the number of data points in the dataset, because then each data point is its own cluster, and there is no error between it and the center of its cluster). So our goal is to choose a small value of k that still has a low SSE, and the elbow usually represents where we start to have diminishing returns by increasing k.

Where k is the number of cluster initially taken i=1.

x is a particular element of cluster i and ci is the corresponding cluster centroid.

d(x, ci) is the error or difference between the cluster elements and centroid.

**5.2 ELBOW method: Obtaining the elbow in the project work**

The project establishes the elbow method i.e. the process to find the minimal no. of clusters among various datasets following both **K-means** and **K-medoids** approach.

**5.2.1 Algorithm:**

**Input:** Taking elements from a file consisting of 2-Dimension data

**Output:** SSE or Sum of Squared Error values are obtained for each cluster (1 to n-1) and taking the outputs to form a chart where we can see the elbow form and the elbow point is considered to be the best number of k.

**Method:**

* + - * 1. After calculating cluster centroid, take the difference or error or‘d’ of every element to the centroid.
        2. Then summing up the errors or‘d’ of each cluster number and storing in SUMi.
        3. If k! =1 then sum up SUMi with SUMi-1 and then with SUMk and store in SSE.
        4. Printing SSE.
        5. Opening Excel Sheet to produce a graph to notice the elbow point.

**5.3.1.1 SSE Value generation and ELBOW graph: K-means**

**EXPERIMENTAL WORK-I**

**Input:**

Elements entered as input**:**

|  |  |
| --- | --- |
| 185 | 72 |
| 170 | 56 |
| 168 | 60 |
| 179 | 68 |
| 182 | 72 |
| 188 | 77 |
| 180 | 71 |
| 180 | 70 |
| 183 | 84 |
| 180 | 88 |
| 180 | 67 |
| 177 | 76 |

**Output generated: SSE value calculated**

|  |  |
| --- | --- |
| **NUMBER OF CLUSTERS** | **SUM OF SQUARED ERROR** |
| **1** | **98.45** |
| **2** | **69.37** |
| **3** | **64.907** |
| **4** | **36.12** |
| **5** | **27.128** |
| **6** | **23.23** |
| **7** | **20.627** |
| **8** | **13.245** |
| **9** | **7.414** |
| **10** | **2.414** |
| **11** | **1.00** |

Table.1.1 SSE values for 12 elements

**Output generated: SSE value plotted to form Elbow Curve**

Fig, 5.1. Elbow Graph: K-means 12

The red point seen on the graph is the **Elbow point** i.e. the best chosen k (number of cluster).

Therefore, the desirable number of cluster in this experiment is 4.

**5.3.1.2 EXPERIMENTAL WORK-II**

**Input:** 240 elements

**Output generated: SSE value calculated**

In this case instead of taking all the 240 numbers of ‘k’, I have chosen the first 10 ‘k’ or the number of cluster so that the elbow graph can have clarity in showing the elbow point.

|  |  |
| --- | --- |
| **NUMBER OF CLUSTERS** | **SUM OF SQUARED ERROR** |
| **1** | **1052.072** |
| **2** | **785.127** |
| **3** | **643.3415** |
| **4** | **490.285** |
| **5** | **451.5178** |
| **6** | **410.459** |
| **7** | **373.8852** |
| **8** | **342.1152** |
| **9** | **326.2282** |
| **10** | **319.0226** |

Table.1.2 SSE values for 240 elements

**Output generated: SSE value plotted to form Elbow Curve**

Fig, 5.2 Elbow Graph: K-means - 240

The red point seen on the graph is the **Elbow point** i.e. the best chosen k (number of cluster).

Therefore, the desirable number of cluster in this experiment is 4.

**5.3.1.3 EXPERIMENTAL WORK-III**

In the work, I have taken a dataset comprised of 500 elements of 2-Dimension nature.

**Input:** 500 elements

**Output generated: SSE value calculated**

In this case instead of taking all the 500 numbers of ‘k’, I have chosen the first 10 ‘k’ or the number of cluster so that the elbow graph can have clarity in showing the elbow point.

|  |  |
| --- | --- |
| **NUMBER OF CLUSTERS** | **SUM OF SQUARED ERROR** |
| **1** | **2630966.484** |
| **2** | **1746431.322** |
| **3** | **1107771.003** |
| **4** | **895474.0886** |
| **5** | **1018454.334** |
| **6** | **805811.0467** |
| **7** | **751575.5183** |
| **8** | **735398.0523** |
| **9** | **678553.4726** |
| **10** | **640443.8868** |

Table.1.3 SSE values for 500 elements

**Output generated: SSE value plotted to form Elbow Curve**

Fig, 5.3 Elbow Graph: K-means - 500

The red points seen on the graph are the **Elbow points,** since there are multiple elbow points so the graph implies ambiguity in finding the best chosen number of clusters needed for proper clustering. However, the desirable number of cluster in this experiment can be either 3 or 5.

**5.3.2.1 SSE Value generation and ELBOW graph: K-medoids**

**EXPERIMENTAL WORK-I**

**Input:**

Same elements entered before taken as input**:**

**Output generated: SSE value calculated**

|  |  |
| --- | --- |
| **NUMBER OF CLUSTERS** | **SUM OF SQUARED ERROR** |
| **1** | **121** |
| **2** | **79** |
| **3** | **51** |
| **4** | **39** |
| **5** | **31** |
| **6** | **23** |
| **7** | **16** |
| **8** | **12** |
| **9** | **6** |
| **10** | **3** |
| **11** | **1** |

Table.2.1 SSE values for 12 elements

**Output generated: SSE value plotted to form Elbow Curve**

Fig, 6.1 Elbow Graph: K-medoids - 12

The red point seen on the graph is the **Elbow point** i.e. the best chosen k (number of cluster).

Therefore, the desirable number of cluster in this experiment is 3.

**5.3.2.2 EXPERIMENTAL WORK-II**

**Input:** 240 elements

**Output generated: SSE value calculated**

In this case instead of taking all the 240 numbers of ‘k’, I have chosen the first 10 ‘k’ or the number of cluster so that the elbow graph can have clarity in showing the elbow point.

|  |  |
| --- | --- |
| **NUMBER OF CLUSTERS** | **SUM OF SQUARED ERROR** |
| **1** | **2630966.484** |
| **2** | **1746431.322** |
| **3** | **1107771.003** |
| **4** | **895474.0886** |
| **5** | **1018454.334** |
| **6** | **805811.0467** |
| **7** | **751575.5183** |
| **8** | **735398.0523** |
| **9** | **678553.4726** |
| **10** | **640443.8868** |

Table.2.2 SSE values for 240 elements

**Output generated: SSE value plotted to form Elbow Curve**

Fig, 6.2 Elbow Graph: K-medoids – 240

The red point seen on the graph is the **Elbow point** i.e. the best chosen k (number of cluster).

Therefore, the desirable number of cluster in this experiment is 4.

**5.3.2.3 EXPERIMENTAL WORK-III**

**Input:** 500 elements

**Output generated: SSE value calculated**

In this case instead of taking all the 500 numbers of ‘k’, I have chosen the first 10 ‘k’ or the number of cluster so that the elbow graph can have clarity in showing the elbow point.

|  |  |
| --- | --- |
| **NUMBER OF CLUSTERS** | **SUM OF SQUARED ERROR** |
| **1** | **3.34E+06** |
| **2** | **2.20E+06** |
| **3** | **1.38E+06** |
| **4** | **1.14E+06** |
| **5** | **1.06E+06** |
| **6** | **1.00E+06** |
| **7** | **927656** |
| **8** | **868344** |
| **9** | **817587** |
| **10** | **761930** |

Table.2.3 SSE values for 500 elements

**Output generated: SSE value plotted to form Elbow Curve**

Fig, 6.3 Elbow Graph: K-medoids - 500

The red point seen on the graph is the **Elbow point** i.e. the best chosen k (number of cluster).

Therefore, the desirable number of cluster in this experiment is 3.

**5.4 SSE Value generation and ELBOW graph: Difference seen**

**in results between K-means and K-medoids**

As it is known that k-medoid is based on centroids (or medoids) calculating by minimizing the absolute distance between the points and the selected centroid, rather than minimizing the square distance. As a result, it's more robust to noise and outliers than k-means. On the other hand, K-means

Hence, the SSE values from K-medoids are somehow found to be lesser than that of K-means since precisely the clusters are formed.

Also, we can see the graph from K-medoids gives a proper elbow point and in case of K-means some (can be negligible) elbow points are shown i.e. ambiguous elbow points are found.

**CHAPTER 6**

**CONCLUSION**

Clustering consist of grouping objects in sets, such that objects within a cluster are as similar as possible, whereas objects from different clusters are as dissimilar as possible. Thus, the optimal clustering is somehow subjective and dependent on the characteristic used for determining similarities, as well as on the level of detail required from the partitions. The estimation of the optimal number of clusters within a set of data points is a very important problem, as most clustering algorithms need that parameter as input in order to group the data. Many methods have been proposed to find the proper, K among which the “elbow” method offers a very clear and naive solution based on intra-cluster variance. But adding to these it is to be kept in mind while performing Elbow method by practicing K-medoids algorithm, I have seen that the time complexity rises high with increasing cluster number. So working with huge datasets can yield result after a prolonged period of time. Also it is seen that with the increasing number of elements in the dataset, the graph shows inability to point out proper Elbow points and also there can be a little ambiguity in finding the point. However, generally the project is successful in figuring out the best possible method to find the minimal number of cluster to form. In this way, this application will definitely help the organization in R&D platform for further research that involves clustering.

**CHAPTER 7**

**LIMITATIONS**

**CHAPTER 8**

**REFERENCES**

* <https://en.wikipedia.org/wiki/Cluster_analysis>
* Data Mining: Concepts and Techniques, Jiawei Han.
* https://sites.google.com/site/dataclusteringalgorithms/k-means-clustering-algorithm
* https://en.wikipedia.org/wiki/K-means\_clustering
* https://en.wikipedia.org/wiki/K-medoids
* https://www.researchgate.net/publication/220215167\_A\_simple\_and\_fast\_algorithm\_for\_K-medoids\_clustering
* www.google.com/images
* www.bing.com/images