Comparison of Euclidean Distance Function and Manhattan Distance Function Using K-Mediods

Md. Mohibullah
Student (M.Sc. - Thesis)
Department of Computer Science
& Engineering
Comilla University
Comilla, Bangladesh

Md. Zakir Hossain
Assistant Professor
Department of Computer Science
& Engineering
Comilla University
Comilla, Bangladesh

Mahmudul Hasan
Assistant Professor
Department of Computer Science
& Engineering
Comilla University
Comilla, Bangladesh.

***Abstract*--Clustering is one kind of unsupervised learning methods. K-mediods is one of the partitioning clustering algorithms and it is also a distance based clustering. Distance measure is an important component of a clustering algorithm to measure the distances between data points. In this thesis paper, a comparison between Euclidean distance function and Manhattan distance function by using K-mediods has been made. To make this comparison, an instance of seven objects of a data set has been taken. Finally, we will show the simulation results in the result section of this paper.**

Keywords-- *Clustering, K-mediods, Manhattan distance function, Euclidean distance function.*

I. INTRODUCTION

Unsupervised learning works on a given set of records (e.g. observations or variables) with no attribute and organize them into groups, without advance knowledge of the definitions of the groups [1]. Clustering is one of the most important unsupervised learning techniques. Clustering, also known as cluster analysis), aims to organize a collection of data items into clusters, such that items within a cluster are more “similar” to each other than they are to items in the other clusters [2]. Clustering methods can be divided into two basic types: hierarchical and partition clustering [3]. There are many partition-based algorithms such as K-Means, K-Mediods and Fuzzy C-Means clustering etc.

The k-means method uses centroid to represent the cluster and it is sensitive to outliers. This means, a data object with an extremely large value may disrupt the distribution of data. K-medoids method overcomes this problem by using medoids to represent the cluster rather than centroid. A medoid is the most centrally located data object in a cluster [4].

II. THE REASON BEHIND CHOOSING K-MEDIODS ALGORITHM

1. K-medoid is more flexible
First of all, k-medoids can be used with any similarity measure. K-means however, may fail to converge - it really must only be used with distances that are consistent with the mean. So e.g. Absolute Pearson Correlation must not be used with k-means, but it works well with k-medoids.

2. Robustness of medoid
Secondly, the medoid as used by k-medoids is roughly comparable to the median. It is a more robust estimate of a representative point than the mean as used in k-means.

III. K-MEDOIDS ALGORITHM (PAM-PARTITIONING AROUND MEDOIDS)

Algorithm [4, 6]
Input
    K: the number of clusters
    D: a data set containing n objects
Output: A set of k clusters.

Method

```
1. Compute distance (cost) so as to associate each
data point to its nearest medoid using Manhattan
distance and/or Euclidean distance.

2.  for each medoid m

    1.  for each non-medoid data point o
            1.  Swap m and o and compute the
                total cost of the configuration

3.  Select the configuration with the lowest cost.

4. Repeat steps 1 to 3 until there is no change in the
medoid.
```

IV. DEMONSTRATION OF K-MEDOIDS

We will see the clustering of data set with an example for k-medoid algorithm using both the Manhattan distance and Euclidean distance.

For Instance: Consider a data set of seven objects as follows:

Serial No	Variable-1	Variable-2
1 (X_1)	1.0	1.0
2 (X_2)	1.5	2.0
3 (X_3)	3.0	4.0
4 (X_4)	5.0	7.0
5 (X_5)	3.5	5.0
6 (X_6)	4.5	5.0
7 (X_7)	3.5	4.5

Table 1: A data set of seven objects

The following shows the scatter diagram of the above data set.

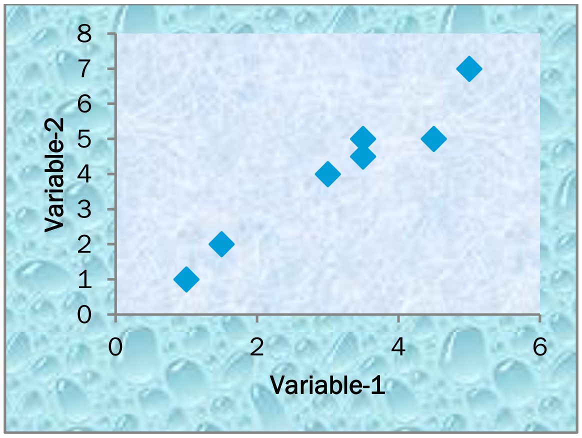

Figure 1: Distribution of data

V. USING MANHATTAN DISTANCE

Step 1

Consider the number of clusters is two i.e., k=2 and initialize *k* centers.

Let us assume $c_1 = (1, 1)$ and $c_2 = (5, 7)$

So here x_1 and x_4 are selected as medoids.

Calculate distance so as to associate each data object to its nearest medoid. Cost is calculated using Manhattan Distance. Costs to the nearest medoid are shown bold in the table.

i	C_1		Data objects (X_i)		Cost (distance)
2	1.0	1.0	1.5	2.0	|1.0 - 1.5|+|1.0 - 2.0|=**1.5**
3	1.0	1.0	3.0	4.0	|1.0 – 3.0|+|1.0 – 4.0|=**5.0**
5	1.0	1.0	3.5	5.0	|1.0 – 3.5|+|1.0 – 5.0|=6.5
6	1.0	1.0	4.5	5.0	|1.0 – 4.5|+|1.0 – 5.0|=7.5
7	1.0	1.0	3.5	4.5	|1.0 – 3.5|+|1.0 – 4.5|=6.0

i	C_2		Data objects (X_i)		Cost (distance)
2	5.0	7.0	1.5	2.0	|5.0 - 1.5|+|7.0 - 2.0|=8.5
3	5.0	7.0	3.0	4.0	|5.0 – 3.0|+|7.0 – 4.0|=**5.0**
5	5.0	7.0	3.5	5.0	|5.0 – 3.5|+|7.0 – 5.0|=**3.5**
6	5.0	7.0	4.5	5.0	|5.0 – 4.5|+|7.0 – 5.0|=**2.5**
7	5.0	7.0	3.5	4.5	|5.0 – 3.5|+|7.0 – 4.5|=**4.0**

Since the cost for X_2 is not changed. So we can keep it in cluster-1. Then the clusters become:

Cluster-1= {(1, 1), (1.5, 2), (3, 4)} i.e. $\{X_1, X_2, X_3\}$

Cluster-2 = {(5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. $\{X_4, X_5, X_6, X_7\}$

Since the points (3.5, 5), (4.5, 5), and (3.5, 4.5) are closer to C_2, hence they form one cluster and the remaining points form another cluster C_1.

So the total cost involved is 16.5.

Where the cost is calculated by following formula:

$$\mathrm{Cost}(X,C)= \sum_{i=0}^{n} | Xi - Ci |$$

Where *x* is any data object, *c* is the medoid, and *n* is the dimension of the object which in this case is 2.

Total cost is the summation of the minimum cost of data object from its medoid in its cluster so here:

Total cost= (1.5+5) + (3.5+2.5+4) =16.5

Step 2

Select one of the nonmedoids O′. Let us assume O′ = (4.5, 5.0). So now the medoids are C_1 (1, 1) and O′ (4.5, 5).

Again, calculate distance so as to associate each data object to its nearest medoid. Costs to the nearest medoid are shown bold in the table.

i	**C_1**		Data objects (X_i)		Cost (distance)
2	1.0	1.0	1.5	2.0	|1.0 - 1.5|+|1.0 - 2.0|=**1.5**
3	1.0	1.0	3.0	4.0	|1.0 – 3.0|+|1.0 – 4.0|=5.0
4	1.0	1.0	5.0	7.0	|1.0 – 5.0|+|1.0 – 7.0|=10
5	1.0	1.0	3.5	5.0	|1.0 – 3.5|+|1.0 – 5.0|=6.5
7	1.0	1.0	3.5	4.5	|1.0 – 3.5|+|1.0 – 4.5|=6.0

i	**O′**		Data objects (X_i)		Cost (distance)
2	4.5	5.0	1.5	2.0	|4.5 - 1.5|+|5.0 - 2.0|=6
3	4.5	5.0	3.0	4.0	|4.5 – 3.0|+|5.0 – 4.0|=**2.5**
4	4.5	5.0	5.0	7.0	|4.5 – 5.0|+|5.0 – 7.0|=**2.5**
5	4.5	5.0	3.5	5.0	|4.5 – 3.5|+|5.0 – 5.0|=**1**
7	4.5	5.0	3.5	4.5	|4.5 – 3.5|+|5.0 – 4.5|=**1.5**

From the step 2, we get the following clusters:

Cluster-1= {(1, 1), (1.5, 2)} i.e. {X_1, X_2}

Cluster-2 = {(3, 4), (5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_3, X_4, X_5, X_6, X_7}

The total cost= 1.5 +2.5 + 2.5 +1+1.5= 9

Cost comparison

From step 1 and step 2, we get the total cost are 16.5 and 9 respectively. So cost of swapping medoid from C_2 to O′ is

S= Current total cost – Past total cost

= 9 – 16.5

= -7.5<0

So moving would be a good idea and the previous choice was a bad idea. Now we will try to again to certain for the clustering.

Step 3

Select another nonmedoid P′. Let us assume P′ = (3.5, 4.5). So now the medoids are C_1 (1, 1) and P′ (3.5, 4.5).

Again, calculate distance so as to associate each data object to its nearest medoid. Cost is calculated using Manhattan Distance. Costs to the nearest medoid are shown bold in the table.

i	**C_1**		Data objects (X_i)		Cost (distance)
2	1.0	1.0	1.5	2.0	|1.0 - 1.5|+|1.0 - 2.0|=**1.5**
3	1.0	1.0	3.0	4.0	|1.0 – 3.0|+|1.0 – 4.0|=5.0
4	1.0	1.0	5.0	7.0	|1.0 – 5.0|+|1.0 – 7.0|=10
5	1.0	1.0	3.5	5.0	|1.0 – 3.5|+|1.0 – 5.0|=6.5
6	1.0	1.0	4.5	5.0	|1.0 – 4.5|+|1.0 – 5.0|=7.5

i	**P′**		Data objects (X_i)		Cost (distance)
2	3.5	4.5	1.5	2.0	|3.5 - 1.5|+|4.5 - 2.0|=4.5
3	3.5	4.5	3.0	4.0	|3.5 – 3.0|+|4.5 – 4.0|=**1.0**
4	3.5	4.5	5.0	7.0	|3.5 – 5.0|+|4.5 – 7.0|=**4.0**
5	3.5	4.5	3.5	5.0	|3.5 – 3.5|+|4.5 – 5.0|=**0.5**
6	3.5	4.5	4.5	5.0	|3.5 – 4.5|+|4.5 – 5.0|=**1.5**

From the step 3, we get the following clusters:

Cluster-1= {(1, 1), (1.5, 2)} i.e. {X_1, X_2}

Cluster-2 = {(3, 4), (5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_3, X_4, X_5, X_6, X_7}

The total cost= 1.5 + 1.0 + 4.0 + 0.5 + 1.5= 8.5

Cost comparison

From step 2 and step 3, we get the total cost are 9.0 and 8.5 respectively. So cost of swapping medoid from O′ to P′ is

S= Current total cost – Past total cost

= 8.5 – 9.0

= -0.5<0

So moving would be a good idea.

Step 4

Select another nonmedoid Q′. Let us assume Q′ = (3.5, 5.0). So now the medoids are C_1 (1, 1) and Q′ (3.5, 5.0).

Again, calculate distance so as to associate each data object to its nearest medoid. Costs to the nearest medoid are shown bold in the table.

i	**C_1**		Data objects (X_i)		Cost (distance)
2	1.0	1.0	1.5	2.0	|1.0 - 1.5|+|1.0 - 2.0|=**1.5**
3	1.0	1.0	3.0	4.0	|1.0 – 3.0|+|1.0 – 4.0|=5.0
4	1.0	1.0	5.0	7.0	|1.0 – 5.0|+|1.0 – 7.0|=10
6	1.0	1.0	4.5	5.0	|1.0 – 4.5|+|1.0 – 5.0|=7.5
7	1.0	1.0	3.5	4.5	|1.0 – 3.5|+|1.0 – 4.5|=6.0

i	**Q′**		Data objects (X_i)		Cost (distance)
2	3.5	5.0	1.5	2.0	|3.5 - 1.5|+|5.0 - 2.0|=5.0
3	3.5	5.0	3.0	4.0	|3.5 – 3.0|+|5.0 – 4.0|=**1.5**
4	3.5	5.0	5.0	7.0	|3.5 – 5.0|+|5.0 – 7.0|=**3.5**
6	3.5	5.0	4.5	5.0	|3.5 – 4.5|+|5.0 – 5.0|=**1.0**
7	3.5	5.0	3.5	4.5	|3.5 – 3.5|+|5.0 – 4.5|=**0.5**

From the step 4, we get the following clusters:

Cluster-1= {(1, 1), (1.5, 2)} i.e. {X_1, X_2}

Cluster-2 = {(3, 4), (5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_3, X_4, X_5, X_6, X_7}

The total cost= 1.5 + 1.5 + 3.5 + 1.0 + 0.5= 8.0

Cost comparison

From step 3 and step 4, we get the total cost are 8.5 and 8.0 respectively. So cost of swapping medoid from P′ to Q′ is

S= Current total cost – Past total cost

= 8.0 – 8.5

= -0.5<0

So moving would be a good idea.

Step 5

Select another nonmedoid R′. Let us assume R′ = (3.0, 4.0). So now the medoids are C_1 (1, 1) and R′ (3.0, 4.0).

Again, calculate distance so as to associate each data object to its nearest medoid. Costs to the nearest medoid are shown bold in the table.

i	**C_1**		Data objects (X_i)		Cost (distance)
2	1.0	1.0	1.5	2.0	|1.0 - 1.5|+|1.0 - 2.0|=**1.5**
4	1.0	1.0	5.0	7.0	|1.0 – 5.0|+|1.0 – 7.0|=10
5	1.0	1.0	3.5	5.0	|1.0 – 3.5|+|1.0 – 5.0|=6.5
6	1.0	1.0	4.5	5.0	|1.0 – 4.5|+|1.0 – 5.0|=7.5
7	1.0	1.0	3.5	4.5	|1.0 – 3.5|+|1.0 – 4.5|=6.0

i	**R′**		Data objects (X_i)		Cost (distance)
2	3.0	4.0	1.5	2.0	|3.0 - 1.5|+|4.0 - 2.0|=3.5
4	3.0	4.0	5.0	7.0	|3.0 – 5.0|+|4.0 – 7.0|=**5.0**
5	3.0	4.0	3.5	5.0	|3.0 – 3.5|+|4.0 – 5.0|=**1.5**
6	3.0	4.0	4.5	5.0	|3.0 – 4.5|+|4.0 – 5.0|=**2.5**
7	3.0	4.0	3.5	4.5	|3.0 – 3.5|+|4.0 – 4.5|=**1.0**

From the step 5, we get the following clusters:

Cluster-1= {(1, 1), (1.5, 2)} i.e. {X_1, X_2}

Cluster-2 = {(3, 4), (5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_3, X_4, X_5, X_6, X_7}

The total cost= 1.5 + 5.0 + 1.5 + 2.5 + 1.0= 11.5

Cost comparison

From step 4 and step 5 we get the total cost are 8.0 and 11.5 respectively. So cost of swapping medoid from Q′ to R′ is

S= Current total cost – Past total cost

= 11.5 – 8.0

= 3.5>0

So moving would be a bad idea and the previous choice was a good idea.

Step 6

Select another nonmedoid S′. Let us assume S′ = (1.5, 2.0). So now the medoids are C_1 (1, 1) and S′ (1.5, 2.0).

Again, calculate distance so as to associate each data object to its nearest medoid. Costs to the nearest medoid are shown bold in the table.

i	C_1		Data objects (X_i)		Cost (distance)
3	1.0	1.0	3.0	4.0	|1.0 – 3.0|+|1.0 – 4.0|=5.0
4	1.0	1.0	5.0	7.0	|1.0 – 5.0|+|1.0 – 7.0|=10
5	1.0	1.0	3.5	5.0	|1.0 – 3.5|+|1.0 – 5.0|=6.5
6	1.0	1.0	4.5	5.0	|1.0 – 4.5|+|1.0 – 5.0|=7.5
7	1.0	1.0	3.5	4.5	|1.0 – 3.5|+|1.0 – 4.5|=6.0

i	S′		Data objects (X_i)		Cost (distance)
3	1.5	2.0	3.0	4.0	|1.5– 3.0|+|2.0– 4.0|=**3.5**
4	1.5	2.0	5.0	7.0	|1.5– 5.0|+|2.0– 7.0|=**8.5**
5	1.5	2.0	3.5	5.0	|1.5– 3.5|+|2.0– 5.0|=**4.5**
6	1.5	2.0	4.5	5.0	|1.5– 4.5|+|2.0– 5.0|=**6.0**
7	1.5	2.0	3.5	4.5	|1.5– 3.5|+|2.0– 4.5|=**4.0**

From the step 6, we get the following clusters:

Cluster-1= {(1, 1)} i.e. {X_1}

Cluster-2 = {(1.5, 2), (3, 4), (5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_2, X_3, X_4, X_5, X_6, X_7}

The total cost= 3.5 + 8.5 + 4.5 + 6.0 + 4.0= 26.5

Cost comparison

From step 4 and step 6 we get the total cost are 8.0 and 26.5 respectively. So cost of swapping medoid from Q′ to S′ is

S= Current total cost – Past total cost

= 26.5 – 8.0

= 18.5 > 0

So moving would be a bad idea and the choice in step 4 was a good idea. So the configuration does not change after step 4 and algorithm terminates here (i.e. there is no change in the medoids- the medoids are X_1 and X_5).

VI. USING EUCLIDEAN DISTANCE

Step 1

Consider the number of clusters is two i.e., k=2 and initialize *k* centers.

Let us assume c_1 = (1, 1) and c_2 = (5, 7)

So here x_1 and x_4 are selected as medoids.

Calculate distance so as to associate each data object to its nearest medoid. Cost is calculated using Euclidean Distance. Costs to the nearest medoid are shown bold in the table.

i	C_1		Data objects (X_i)		Cost (distance)
2	1.0	1.0	1.5	2.0	$\sqrt{\|1.0-1.5\|^2+\|1.0-2.0\|^2} = \mathbf{1.118}$
3	1.0	1.0	3.0	4.0	$\sqrt{\|1.0-3.0\|^2+\|1.0-4.0\|^2} = \mathbf{3.606}$
5	1.0	1.0	3.5	5.0	$\sqrt{\|1.0-3.5\|^2+\|1.0-5.0\|^2} = 4.717$
6	1.0	1.0	4.5	5.0	$\sqrt{\|1.0-4.5\|^2+\|1.0-5.0\|^2} = 5.315$
7	1.0	1.0	3.5	4.5	$\sqrt{\|1.0-3.5\|^2+\|1.0-4.5\|^2} = 4.301$

i	C_2		Data objects (X_i)		Cost (distance)
2	5.0	7.0	1.5	2.0	$\sqrt{\|5.0-1.5\|^2+\|7.0-2.0\|^2} = 6.103$
3	5.0	7.0	3.0	4.0	$\sqrt{\|5.0-3.0\|^2+\|7.0-4.0\|^2} = \mathbf{3.606}$
5	5.0	7.0	3.5	5.0	$\sqrt{\|5.0-3.5\|^2+\|7.0-5.0\|^2} = \mathbf{2.5}$

6	5.0	7.0	4.5	5.0	$\sqrt{\|5.0-4.5\|^2+\|7.0-5.0\|^2}$ $= \mathbf{2.062}$
7	5.0	7.0	3.5	4.5	$\sqrt{\|5.0-3.5\|^2+\|7.0-4.5\|^2}$ $= \mathbf{2.915}$

Since the cost for X_2 is not changed. So we can keep it in cluster-1. Then the clusters become:

Cluster-1= {(1, 1), (1.5, 2), (3, 4)} i.e. {X_1, X_2, X_3}

Cluster-2 = {(5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_4, X_5, X_6, X_7}

Since the points (3.5, 5), (4.5, 5), and (3.5, 4.5) are closer to C_2, hence they form one cluster and the remaining points form another cluster C_1.

Total cost is the summation of the minimum cost of data object from its medoid in its cluster so here:

Total cost= (1.118+3.606) + (2.5+2.062+2.915) =12.201

Step 2

Select one of the nonmedoids O′. Let us assume O′ = (4.5, 5.0). So now the medoids are C_1 (1, 1) and O′ (4.5, 5).

Again, calculate distance so as to associate each data object to its nearest medoid. Costs to the nearest medoid are shown bold in the table.

i	**C₁**		Data objects (X_i)		Cost (distance)
2	1.0	1.0	1.5	2.0	$\sqrt{\|1.0-1.5\|^2+\|1.0-2.0\|^2}$ $= \mathbf{1.118}$
3	1.0	1.0	3.0	4.0	$\sqrt{\|1.0-3.0\|^2+\|1.0-4.0\|^2}$ $= 3.606$
4	1.0	1.0	5.0	7.0	$\sqrt{\|1.0-5.0\|^2+\|1.0-7.0\|^2}$ $= 7.211$
5	1.0	1.0	3.5	5.0	$\sqrt{\|1.0-3.5\|^2+\|1.0-5.0\|^2}$ $= 4.717$
7	1.0	1.0	3.5	4.5	$\sqrt{\|1.0-3.5\|^2+\|1.0-4.5\|^2}$ $= 4.301$

i	O′		Data objects (X_i)		Cost (distance)
2	4.5	5.0	1.5	2.0	$\sqrt{\|4.5-1.5\|^2+\|5.0-2.0\|^2}$ $= 4.243$
3	4.5	5.0	3.0	4.0	$\sqrt{\|4.5-3.0\|^2+\|5.0-4.0\|^2}$ $= \mathbf{1.803}$
4	1.0	1.0	5.0	7.0	$\sqrt{\|4.5-5.0\|^2+\|5.0-7.0\|^2}$ $= \mathbf{2.062}$
5	4.5	5.0	3.5	5.0	$\sqrt{\|4.5-3.5\|^2+\|5.0-5.0\|^2}$ $= \mathbf{1.0}$
7	4.5	5.0	3.5	4.5	$\sqrt{\|4.5-3.5\|^2+\|5.0-4.5\|^2}$ $= \mathbf{1.118}$

From the step 2, we get the following clusters:

Cluster-1= {(1, 1), (1.5, 2)} i.e. {X_1, X_2}

Cluster-2 = {(3, 4), (5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_3, X_4, X_5, X_6, X_7}

The total cost= 1.118 +1.803 + 2.062 +1.0+1.118= 7.101

Cost comparison

From step 1 and step 2, we get the total cost are 12.201 and 7.101 respectively. So cost of swapping medoid from C_2 to O′ is

S= Current total cost – Past total cost

= 7.101 – 12.201

= -5.1<0

So moving would be a good idea and the previous choice was a bad idea. Now we will try to again to certain for the clustering.

Step 3

Select another nonmedoid P′. Let us assume P′ = (3.5, 4.5). So now the medoids are C_1 (1, 1) and P′ (3.5, 4.5).

Again, calculate distance so as to associate each data object to its nearest medoid. Costs to the nearest medoid are shown bold in the table.

i	**C₁**	Data objects (X_i)	Cost (distance)

2	1.0	1.0	1.5	2.0	$\sqrt{\|1.0-1.5\|^2+\|1.0-2.0\|^2}$ $= \mathbf{1.118}$
3	1.0	1.0	3.0	4.0	$\sqrt{\|1.0-3.0\|^2+\|1.0-4.0\|^2}$ $= 3.606$
4	1.0	1.0	5.0	7.0	$\sqrt{\|1.0-5.0\|^2+\|1.0-7.0\|^2}$ $= 7.211$
5	1.0	1.0	3.5	5.0	$\sqrt{\|1.0-3.5\|^2+\|1.0-5.0\|^2}$ $= 4.717$
6	1.0	1.0	4.5	5.0	$\sqrt{\|1.0-4.5\|^2+\|1.0-5.0\|^2}$ $= 5.315$

i	P′		Data objects (X_i)		Cost (distance)
2	3.5	4.5	1.5	2.0	$\sqrt{\|3.5-1.5\|^2+\|4.5-2.0\|^2}$ $= 3.202$
3	3.5	4.5	3.0	4.0	$\sqrt{\|3.5-3.0\|^2+\|4.5-4.0\|^2}$ $= \mathbf{0.707}$
4	3.5	4.5	5.0	7.0	$\sqrt{\|3.5-5.0\|^2+\|4.5-7.0\|^2}$ $= \mathbf{2.915}$
5	3.5	4.5	3.5	5.0	$\sqrt{\|3.5-3.5\|^2+\|4.5-5.0\|^2}$ $= \mathbf{0.5}$
6	3.5	4.5	4.5	5.0	$\sqrt{\|3.5-4.5\|^2+\|4.5-5.0\|^2}$ $= \mathbf{1.118}$

From the step 3, we get the following clusters:

Cluster-1= {(1, 1), (1.5, 2)} i.e. {X_1, X_2}

Cluster-2 = {(3, 4), (5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_3, X_4, X_5, X_6, X_7}

The total cost= 1.118 +0.707 + 2.915 +0.5+1.118= 6.358

Cost comparison

From step 2 and step 3, we get the total cost are 7.101 and 6.358 respectively. So cost of swapping medoid from O′ to P ′is

S= Current total cost – Past total cost

= 6.358 – 7.101

= -0.743<0

So moving would be a good idea.

Step 4

Select another nonmedoid Q′. Let us assume Q′ = (3.5, 5). So now the medoids are C₁ (1, 1) and Q′ (3.5, 5).

Again, calculate distance so as to associate each data object to its nearest medoid. Costs to the nearest medoid are shown bold in the table.

i	**C₁**		Data objects (X_i)		Cost (distance)
2	1.0	1.0	1.5	2.0	$\sqrt{\|1.0-1.5\|^2+\|1.0-2.0\|^2}$ $= \mathbf{1.118}$
3	1.0	1.0	3.0	4.0	$\sqrt{\|1.0-3.0\|^2+\|1.0-4.0\|^2}$ $= 3.606$
4	1.0	1.0	5.0	7.0	$\sqrt{\|1.0-5.0\|^2+\|1.0-7.0\|^2}$ $= 7.211$
6	1.0	1.0	4.5	5.0	$\sqrt{\|1.0-4.5\|^2+\|1.0-5.0\|^2}$ $= 5.315$
7	1.0	1.0	3.5	4.5	$\sqrt{\|1.0-3.5\|^2+\|1.0-4.5\|^2}$ $= 4.301$

i	Q′		Data objects (X_i)		Cost (distance)
2	3.5	5.0	1.5	2.0	$\sqrt{\|3.5-1.5\|^2+\|5.0-2.0\|^2}$ $= 3.606$
3	3.5	5.0	3.0	4.0	$\sqrt{\|3.5-3.0\|^2+\|5.0-4.0\|^2}$ $= \mathbf{1.118}$
4	3.5	5.0	5.0	7.0	$\sqrt{\|3.5-5.0\|^2+\|5.0-7.0\|^2}$ $= \mathbf{2.50}$
6	3.5	5.0	4.5	5.0	$\sqrt{\|3.5-4.5\|^2+\|5.0-5.0\|^2}$ $= \mathbf{1.00}$
7	3.5	5.0	3.5	4.5	$\sqrt{\|3.5-3.5\|^2+\|5.0-4.5\|^2}$ $= \mathbf{0.50}$

From the step 4, we get the following clusters:

Cluster-1= {(1, 1), (1.5, 2)} i.e. {X_1, X_2}

Cluster-2 = {(3, 4), (5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_3, X_4, X_5, X_6, X_7}

The total cost= 1.118 +1.118 + 2.5 +1.0+0.5= 6.236

Cost comparison

From step 3 and step 4, we get the total cost are 6.358 and 6.236 respectively. So cost of swapping medoid from P′ to Q ′is

S= Current total cost – Past total cost

= 6.236 – 6.358

= -0.122<0

So moving would be a good idea.

Step 5

Select another nonmedoid R′. Let us assume R′ = (3.0, 4.0). So now the medoids are C_1 (1, 1) and R′ (3.0, 4.0).

Again, calculate distance so as to associate each data object to its nearest medoid. Costs to the nearest medoid are shown bold in the table.

i	C_1		Data objects (X_i)		Cost (distance)
2	1.0	1.0	1.5	2.0	$\sqrt{\|1.0-1.5\|^2+\|1.0-2.0\|^2} = \mathbf{1.118}$
4	1.0	1.0	5.0	7.0	$\sqrt{\|1.0-5.0\|^2+\|1.0-7.0\|^2} = 7.211$
5	1.0	1.0	3.5	5.0	$\sqrt{\|1.0-3.5\|^2+\|1.0-5.0\|^2} = 4.717$
6	1.0	1.0	4.5	5.0	$\sqrt{\|1.0-4.5\|^2+\|1.0-5.0\|^2} = 5.315$
7	1.0	1.0	3.5	4.5	$\sqrt{\|1.0-3.5\|^2+\|1.0-4.5\|^2} = 4.301$

i	R′		Data objects (X_i)		Cost (distance)
2	3.0	4.0	1.5	2.0	$\sqrt{\|3.0-1.5\|^2+\|4.0-2.0\|^2} = 2.5$
4	3.0	4.0	5.0	7.0	$\sqrt{\|3.0-5.0\|^2+\|4.0-7.0\|^2} = \mathbf{3.606}$
5	3.0	4.0	3.5	5.0	$\sqrt{\|3.0-3.5\|^2+\|4.0-5.0\|^2} = \mathbf{1.118}$
6	3.0	4.0	4.5	5.0	$\sqrt{\|3.0-4.5\|^2+\|4.0-5.0\|^2} = \mathbf{1.803}$
7	3.0	4.0	3.5	4.5	$\sqrt{\|3.0-3.5\|^2+\|4.0-4.5\|^2} = \mathbf{0.707}$

From the step 5, we get the following clusters:

Cluster-1= {(1, 1), (1.5, 2)} i.e. {X_1, X_2}

Cluster-2 = {(3, 4), (5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_3, X_4, X_5, X_6, X_7}

The total cost= 1.118 +3.606 + 1.118 +1.803+0.707= 8.352

Cost comparison

From step 4 and step 5 we get the total cost are 6.236 and 8.352 respectively. So cost of swapping medoid from Q′ to R′ is

S= Current total cost – Past total cost

= 8.352 – 6.236

= 2.116>0

So moving would be a bad idea and the previous choice was a good idea.

Step 6

Select another nonmedoid S′. Let us assume S′ = (1.5, 2.0). So now the medoids are C_1 (1, 1) and S′ (1.5, 2.0).

Again, calculate distance so as to associate each data object to its nearest medoid. Costs to the nearest medoid are shown bold in the table.

i	C_1		Data objects (X_i)		Cost (distance)
3	1.0	1.0	3.0	4.0	$\sqrt{\|1.0-3.0\|^2+\|1.0-4.0\|^2} = 3.606$
4	1.0	1.0	5.0	7.0	$\sqrt{\|1.0-5.0\|^2+\|1.0-7.0\|^2} = 7.211$
5	1.0	1.0	3.5	5.0	$\sqrt{\|1.0-3.5\|^2+\|1.0-5.0\|^2} = 4.717$
6	1.0	1.0	4.5	5.0	$\sqrt{\|1.0-4.5\|^2+\|1.0-5.0\|^2} = 5.315$
7	1.0	1.0	3.5	4.5	$\sqrt{\|1.0-3.5\|^2+\|1.0-4.5\|^2} = 4.301$

i	S′		Data objects (X_i)		Cost (distance)
3	1.5	2.0	3.0	4.0	$\sqrt{\|1.5-3.0\|^2+\|2.0-4.0\|^2} = \mathbf{2.50}$

4	1.5	2.0	5.0	7.0	$\sqrt{\lvert 1.5-5.0\rvert^2+\lvert 2.0-7.0\rvert^2} = \mathbf{6.103}$
5	1.5	2.0	3.5	5.0	$\sqrt{\lvert 1.5-3.5\rvert^2+\lvert 2.0-5.0\rvert^2} = \mathbf{3.606}$
6	1.5	2.0	4.5	5.0	$\sqrt{\lvert 1.5-4.5\rvert^2+\lvert 2.0-5.0\rvert^2} = \mathbf{4.243}$
7	1.5	2.0	3.5	4.5	$\sqrt{\lvert 1.5-3.5\rvert^2+\lvert 2.0-4.5\rvert^2} = \mathbf{3.202}$

From the step 6, we get the following clusters:

Cluster-1= {(1, 1)} i.e. {X_1}

Cluster-2 = {(1.5, 2), (3, 4), (5, 7), (3.5, 5), (4.5, 5), (3.5, 4.5)} i.e. {X_2, X_3, X_4, X_5, X_6, X_7}

The total cost= 2.5 +6.103+ 3.606 +4.243+3.202= 19.654

Cost comparison

From step 4 and step 6 we get the total cost are 6.236 and 19.654 respectively. So cost of swapping medoid from Q′ to S′ is

S= Current total cost – Past total cost

= 19.654 – 6.234

= 13.42 > 0

So moving would be a bad idea and the choice in step 4 was a good idea. So the configuration does not change after step 4 and algorithm terminates here (i.e. there is no change in the medoids- the medoids are X_1 and X_5.

VII. COMPARISON RESULTS OF MANHATTAN AND EUCLIDEAN DISTANCE FUNCTION

From the both methods we have seen that the set of clusters are the same and the centroids are X_1 and X_5.

The following figure is the final graphical diagram for our example that is shown in both steps 4.

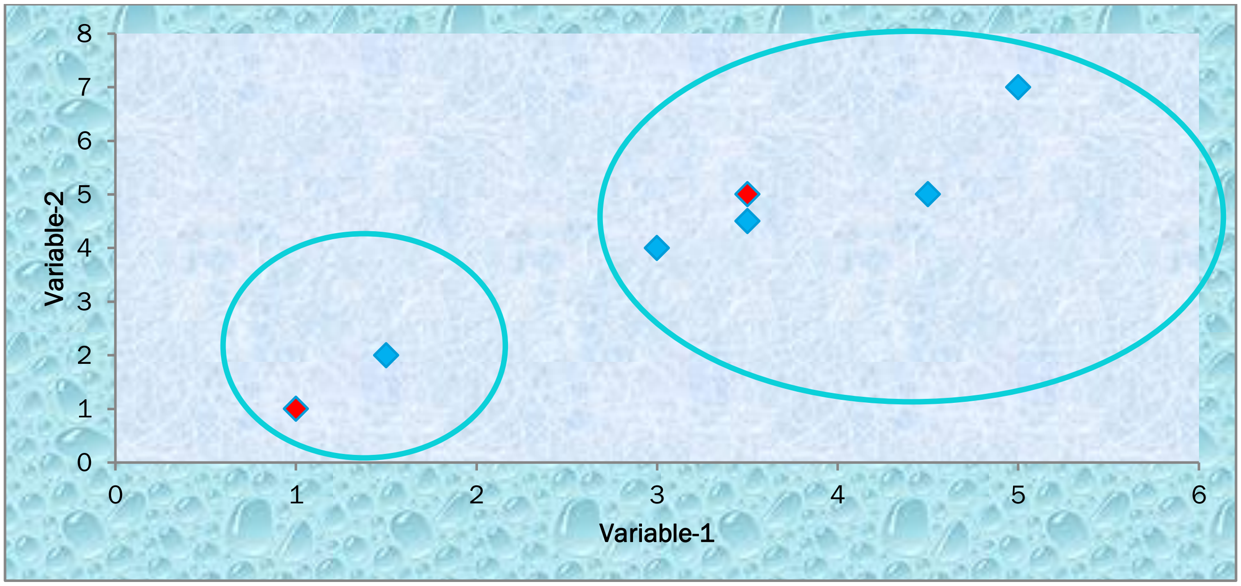

Figure 2: Graphical Diagram of the resultant clustering.

The following figure is the cost function bar-chart diagram for both Manhattan and Euclidean distance.

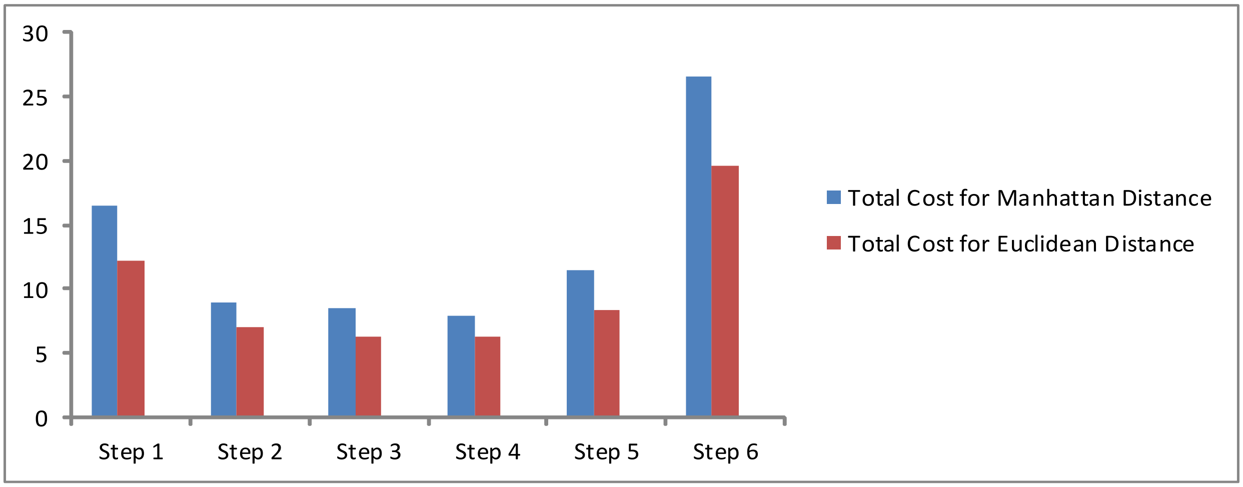

Figure 3: Cost function diagram

We have seen that from the above cost function diagram, the total cost in each step for Euclidean distance is less than the total cost for Manhattan distance. For instance, the total cost in step 1 for Euclidean distance is 12.201 whereas the total cost in the same step for Manhattan distance is 16.5.

CONCLUSION AND FUTURE WORKS

Both Manhattan distance function and Euclidean distance function can be used to cluster of data set for the k-medoid. The Manhattan distance is based on absolute value distance, as opposed to squared error (Euclidean) distance. In practice, you should get similar results most of the time. Although absolute value distance should give more robust results, Euclidean distance function is very effective for small amounts of quality data, and thus favor squared error methods with their greater efficiency.

From the comparison result we can deduce that, Euclidean distance function is really effective for a small set of data. In this paper, we have also seen that, the cost function of each step our given example for the k-medoid method using Euclidean distance function is relatively less than the cost function of corresponding step using Manhattan distance function.

In future, I will work on big data set to cluster effectively.

REFERENCES

[1] Salissou Moutari, Unsupervised learning: Clustering, Centre for Statistical Science and Operational Research (CenSSOR), Queen's University, 17th September 2013.

[2] Nizar Grira, Michel Crucianu, Nozha Boujemaa. Unsupervised and Semi-supervised Clustering: a Brief Survey in A Review of Machine Learning Techniques for Processing Multimedia Content, Report of the MUSCLE European Network of Excellence (6th Framework Programme), August 15, 2005.

[3] http://users.ics.aalto.fi/sami/thesis/node9.html

[4] Shalini S Singh, N C Chauhan. K-means v/s K-medoids: A Comparative Study. National Conference on Recent Trends in Engineering & Technology, 13-14 May 2011.

[5] Amit Singla, Mr. Karambir. Comparative Analysis & Evaluation of Euclidean Distance Function and Manhattan Distance Function Using K-means Algorithm. National Conference on Recent Trends in Engineering & Technology, 13-14 May 2011. IJARCSSE, Volume 2, Issue 7, July 2012.

[6] http://en.wikipedia.org/wiki/K-medoids

[7] Deepak Sinwar, Rahul Kaushik. Study of Euclidean and Manhattan DistanceMetrics using Simple K-Means Clustering. International Journal for

Research in Applied Science and Engineering Technology (IJRASET) ISSN: 2321-9653, Vol. 2 Issue V, May 2014.

[8] J. Han, M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publisher, SanFrancisco, USA, 2001.

[9] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. From Data Mining to Knowledge Discovery in Databases. American Association for Artificial Intelligence, 1996.

[10] Rahmat Widia Sembiring, Jasni Mohamad Zain, Abdullah Embong. Clustering High Dimensional Data Using Subspace and Projected Clustering Algorithms. International journal of computer science & information Technology (IJCSIT) Vol.2, No.4, August 2010.

[11] Isabelle Guyon, André Elisseeff. An Introduction to Variable and Feature Selection. Journal of Machine Learning Research 3 (2003) 1157-1182.

[12] Charu C. Aggarwal, Jiawei Han, Jianyong Wang, Philip S. Y u. A Framework for Projected Clustering of HighDimensional Data Streams. Proceedings of the 30th VLDB Conference, Toronto, Canada, 2004.

[13] A.K. Jain, M.N. Murty, P.J. Flynn. Data Clustering: A Review. ACM Computing Surveys, Vol. 31, No. 3, September 1999.

[14] Man Lung Yiu and Nikos Mamoulis. Iterative Projected Clustering by Subspace Mining. IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 2, February 2005.

[15] Jiawei Han and Micheline Kamber. Data Mining: Concepts and Techniques, Second Edition.

AUTHORS PROFILE

1. **Md. Mohibullah** who obtained B.Sc. (Engg.) in Computer Science and Engineering Department at Comilla University, Comilla, Bangladesh. He is now a student of M.Sc. (Thesis) at this university and a member (student) of Bangladesh Computer Society (BCS). His research interest includes data mining, Artificial Intelligent and Robotics.
2. **Md. Zakir Hossain** is now working as Assistant Professor in the Dept. of Computer Science & Engineering at Comilla University, Bangladesh. He was also a former faculty member of Stamford University Bangladesh in the Dept. of Computer Science & Engineering. He obtained MSc and BSc in Computer Science & Engineering from Jahangirnagar University in 2010 & 2008 respectively. His research interest includes Natural Language Processing, Image Processing, Artificial Intelligent and Software Engineering.
3. **Mahmudul Hasan** who obtained an M.Sc. (Thesis) in Computer Science and Engineering from University of Rajshahi, Bangladesh in 2010, is currently employed as Assistant Professor in the Department of Computer Science and Engineering(CSE) at Comilla University, Comilla, Bangladesh. He worked as a Lecturer at Daffodil International University and Dhaka International University in Dhaka, Bangladesh. His teaching experience includes four under graduate courses, as well as five years of research experience at University of Rajshahi, Bangladesh. He is a member of IAENG (International Association of Engineers). His research activities involve Speech Processing, Bio-Informatics, Networking, and Cryptography.