UMYU Scientifica, Vol. 1 NO. 1, September 2022, Pp 12 – 19
https://doi.org/10.56919/usci.1122.003
ISSN: 2955 – 1145 (print); 2955 – 1153 (online)
A periodical of the Faculty of Natural and Applied Sciences, UMYU, Katsina

ORIGINAL RESEARCH ARTICLE

About Some Data Precaution Techniques For K-Means Clustering Algorithm

Muazu Zulkifilu and Abdulkadir Yasir

Department of Mathematics and Statistics, Umaru Musa Yar'adua University, Katsina, P.M.B 2218 Katsina-Nigeria

ABSTRACT

Clustering is a technique of creating groups of objects such that each group contains similar and unique objects. One of the most popular clustering techniques is the k-means clustering algorithm. Conventional k-means techniques may not work well for high-dimensional datasets, due to the noise, discrepancies, and outliers associated with the original dataset. However, some form of transformation is required to organize the data for clustering. Four different data pre-processing methods are applied before the clustering algorithm to make the data clean, noise-free and consistent. The impact of data pre-processing on the basic k-means clustering algorithm was tested on real-life data using some normalization techniques such as z-score, mean-max, decimal scaling, and mean absolute deviation. We find that the pre-processing before clustering yields good clustering results and significantly reduces the running time compared to the traditional techniques. We can also conclude that the mean absolute deviation is the best among the four normalization methods as it captures all points of clustering.

ARTICLE HISTORY

Received July 29, 2022
Accepted August 25, 2022
Published September 30, 2022

KEYWORDS

K-means clustering, *z*-score, min-max, decimal scale, mean absolute deviation, standardization

INTRODUCTION

K-means is one of the most famous clustering algorithms, data clustering, sometimes called cluster analysis, is a technique for creating groups of objects, such that each cluster contains similar objects and unique (Guojun *et al*., 2007).

Clustering methods mainly focus on pattern recognition for further analysis of organizational design, finding groups of data objects where objects in one group are similar to each other and differ from objects in other groups. Although k-means is one of the most popular clustering algorithms, it has some major drawbacks such as convergence to local minima and initializing the number of clusters containing noise, discrepancies, and outliers in the original dataset.

Cluster analysis is common in all fields related to multivariate data set analysis, and k-means clustering algorithms are among the most popular methods for clustering multivariate observations (Tsai and Chiu, 2008). It is a commonly used system for direct segmentation of aggregates data into k groups. The k-means algorithm produces a quick and efficient solution. The basic k-means algorithm works with the goal of minimizing the mean squared distance between each data point and its nearest center.

The cluster analysis groups objects (observations) based on information found in the data that describes the objects or their relationships (Manimekalai. *et al*,. 2013). The goal is for objects in one group to be similar (or related) to each other and different (or unrelated) to objects in another group. The greater the similarity (or homogeneity) within a cluster, and the greater the difference between clusters, the better or more distinct the clusters.

The purpose of clustering is to find commonalities and designs from large data sets by dividing the data into groups. Since it is assumed that the data set is unlabelled, clustering is often considered as the most valuable unsupervised learning problem (Cios *et al*., 2007).

To get the optimal solution for k-means clustering, the data should be pre-processed before clustering analysis (Chandrasekhar *et al*., 2011).

This pre-processing includes data normalization, princi-

Correspondence: Muazu, Z.; Department of Mathematics and Statistics, Umaru Musa Yar'adua University, Katsina, P.M.B 2218 Katsina-Nigeria. ✉ zmuazufuntua@gmail.com

How to cite: Muazu, Z. and Abdulkadir, Y. (2022). About Some Data Precaution Techniques For K-Means Clustering Algorithm. UMYU Scientifica, 1(1), 12 –19. https://doi.org/10.47430/usci.1122.003

pal component analysis (PCA), single value decomposition (SVD) and others, all of which are intended to detect and remove exceptions. Distinct are points in the given data that are far from the rest in quantity and if not detected and handled correctly; clustering results will be greatly affected (Sairam *et al*., 2012).

Pre-processing (Alshalabi *et al*., 2006) is really necessary before using data mining algorithms to improve the performance of the results. Data set normalization is a part of pre-processing in data mining, where attribute data is scaled to fall within a specified small range. Normalization before clustering is especially necessary for distance metrics, such as Euclidean distance, that are sensitive to variations in magnitude or attribute scale. In real-world applications, due to differences in attribute value selection, one property may override another.

Therefore, it is important to pre-process the data prior to clustering algorithms due to the noise, discrepancies, and outliers associated with the original dataset.

One approach to dealing with outliers is data normalization. This method rescales the dataset to fit within the specified range of values so that attributes with higher values do not dominate attributes with lower values. Normalization is an important pre-processing step in data clustering to normalize the values of all variables from dynamic to specific ranges (Atomi, 2012).

There are no generally defined rules for data set normalization and the choice of a particular normalization rule is therefore largely user-determined (Karthikeyani and Thangavel, 2009). Some data normalization methods include z-score, min-max, decimal scaling and mean absolute deviation.

In z-score, the values of attribute X are normalized to the mean and standard deviation of X, this method is useful when the actual minimum and maximum values of the attribute X are unknown. Decimal scale, the normalized decimal by moving the decimal point of the X attribute values, the number of decimal points moved depends on the maximum absolute value of X. Min-max transforms the data set from 0.0 to 0.1 by subtracting the smallest value for each value divided by the range of values of each individual value. The mean absolute deviation of an item in a data set is the absolute difference between each observation and the mean of the dataset

MATERIALS AND METHODS

Organizing the data for clustering requires some form of transformation, such as normalization, principal component analysis, or single value decomposition (Hans-Joachim *et al.* 2008). In this research we employed some normalization method.

Let Y = {X1, X2,…, Xn} be a d-dimensional raw data set. Then, the data matrix is an n × d matrix given by:

$$(X_1, X_2, X_3, \dots \dots X_n) = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1d} \\ x_{21} & x_{22} & \dots & x_{2d} \\ \vdots & \vdots & \dots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nd} \end{bmatrix} \quad 1$$

Experiments were performed using four different normalization methods: z-score, decimal scaling, min-max, and mean absolute deviation. It aims to test the performance using the basic k-means clustering algorithm and to choose the appropriate data normalization method from these four methods. The sum of squared errors, representing the distance between data points and their cluster centers, was used to measure cluster quality under four normalization approaches.

Z-score

Z-score is a form of normalization technique used to convert normal variations into standardized scoring. For a raw data set Y, the formula for normalizing the Z score is defined as

$$Z(X_i) = \frac{x_i - \mu_i}{\delta_i} \quad 2$$

where x_i and δ_i are the sample mean and standard deviation of the i^{th}attribute respectively. The transformed variable will have a mean of 0 and a variance of 1. Information about the position and scale of the original variable has been lost (Jain and Dubes, 1988). An important limitation of z-score normalization is that it must be applied in global normalization and not in internal normalization (Milligan and Cooper, 1988).

Min-max:

Min-Max normalization is the process of taking data measured in its engineering units and transforming it to a value between (0.0 -1.0), where by the lowest (min) value is set equal to 0.0 and the highest (max) value is set equal to 1.0 respectively. This provides an easy way to compare values that are measured using different scales or different units of measurement. The normalized value is defined as

$$MM(X_{ij}) = \frac{X_{ij} - X_{min}}{X_{max} - X_{min}} \quad 3$$

Decimal scaling

Normalization by decimal scaling normalizes by moving the decimal point of values of feature X, where the number of decimal points moved depends on the maximum absolute value of X. A modified value DS (X) corresponding to X is given by

$$DS(X_{ij}) = \frac{X_{ij}}{10^c} \quad 4$$

Where c is the smallest integer such that

$$max[|DS(X_{ij})|] < 1$$

Mean absolute deviation:

The Mean absolute deviation of an element of a data set is the absolute difference between that element and a given point. Typically, the deviation is reckoned from the central value being construed as some type of average, most often the median or sometimes the mean of the data set.

$$\text{MAD} = \frac{1}{n}\sum_{i=1}^{n}|x_i - m(X)| \quad 5$$

where n is the number of obsevations, X_i is the data element.

***K*-means Cluster Analysis**

The k-means algorithm always converges to a local minimum (Telgarsky and Vattani, 2010). The specific local minimum found depends on the clustering method (Manpreet and Usvir, 2013). The k-means algorithm performs an iterative update of the cluster centers until the local minimum is reached. Before the k-means algorithm converged, the distance and centroid (centre) calculations were performed many times in a loop.. The calculation procedure of the o(nkl) algorithm is very complicated, where n is the total number of objects in the data set, k is the number of clusters required, and l is the number of iterations. The time complexity for the high dimensional dataset is o(nmkl), where m is the number of dimensions.

Given a set of observations, $X = (X_1, X_2, \dots, X_n)$ where each observation is a p-dimensional real vector, k-means clustering aims to divide the n observations into k sets $(k \leq n)$,

$G = (g_1, g_2, \dots, g_k)$ to minimize the sum of squares within the cluster (SSWC).

$$arg_g min \sum_{j=1}^{k} \sum_{x_j \epsilon C_j} ||X_j - \mu_j||^2 \qquad 6$$

where $\mu_j = \frac{1}{n}\sum_{X_j \epsilon C_j} X_j$ denotes the centroid of a cluster c_j.

The weakness of this algorithm is that it can converge to a local minimum of the value of the criterion function if the original data is not pre-processed correctly. The local minimum is the minimum value in the set of points that may or may not be a common minimum and is not the lowest value in the set. Its computation time is also very high, especially for large datasets. Therefore, to get the optimal solution for k-means cluster analysis, the data need to be pre-processed before k-means cluster analysis (Chandrasekhar *et al*., 2011).

Basic *K*-means Method

The steps in the basic k-means method are to scale the data to fall within a specific range of values such that no variable with a larger domain overwhelms a variable with a smaller domain using four methods: z -score, min- max, decimal rate and mean absolute deviation. Then, the reduced data set obtained will be applied to the k-means clustering algorithm and the method that gives the smallest total error of a square and captures all points in the cluster formation will be considered the best method among others. The steps of the technique are as follows:

1st Step

Consider four methods of normalizing data: z-score, min-max, decimal scaling, and mean absolute deviation. For convenience, For convenience, let $X = (X_1', X_2', \dots, X_n')$ is the d-dimensional raw dataset. This results in an $n \times d$ data matrix given by

$$X = (X_1', X_2', \dots, X_n') = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1d} \\ x_{21} & x_{22} & \dots & x_{2d} \\ \vdots & \vdots & \dots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nd} \end{bmatrix} \qquad 7$$

The z-score of X_i is given as

$$Z(X_i) = \frac{x_{ij} - \mu_i}{\delta_i} \qquad 8$$

Where x_{ij}'s are the normalized raw score, μ and σ are the population mean and population standard deviation of the dataset respectively. Since both values are unknown, they will be represented by the sample mean $\bar{x}$ and sample standard deviation s.

The min-max of X_i is given by

$$MM(X_{ij}) = \frac{X_{ij} - X_{min}}{X_{max} - X_{min}} \qquad 9$$

where the lowest (minimum) value is set equal to 0.0 and the highest (maximum) value is set equal to 1.0 respectively.

The Decimal scaling of X_i is given by

$$DS(X_{ij}) = \frac{X_{ij}}{10^c} \qquad 10$$

Where 10^c is the smallest integer such that $max[|DS(X_{ij})|] < 1$

The Mean absolute deviation of X_i is given by

$$\text{MAD} = \frac{1}{n}\sum_{i=1}^{n} |x_i - m(X)| \qquad 11$$

where n is the number of observation, X_i is the row vector

2nd Step

Consider a set of observations $X = (X_1', X_2', \dots, X_n')$ where each observation is a real vector of p-dimensions. To divide the observations into k sets $(k \leq n)$, $G = (g_1, g_2, \dots, g_k)$, Calculate

$$d_{euclidean}(XY) = \sqrt{(x_i - y_i)^2} = [(x-y)(x-y)']^{\frac{1}{2}}. \qquad 12$$

Where X and Y are m-dimensional vectors and denoted as $X = (x_1, x_2, x_3, \cdots, x_m) \varepsilon \mathbb{R}^m$

And $Y = (y_1, y_2, y_3, \cdots, y_m) \varepsilon \mathbb{R}^m$ represent m attribute values of two records (Larose, 2005).

The algorithm proceeds by alternating between two steps

$$G_i = \{x_p : ||x_p - \mu_i||^2 \leq ||x_p - \mu_j||^2 \; \forall \, j, 1 \leq j \leq k\} \qquad 13$$

Where each x_p is assign to exactly one G, then update the process by computing new centers in the new clusters. The algorithm converges when this assignment no longer changes. Then calculate the total sum of squares error (SSE) as

$$SSE = argmin \sum_{i=1}^{k} \sum_{j=1}^{p} \|x_{ij} - \mu_i\|^2 \qquad 14$$

where

$\mu_i = \frac{1}{p}\sum_{j=1}^{p} X_{ij}$ Represents the centre of a cluster and p represents the number of individuals.

RESULTS AND DISCUSSION

Here the technical method of data preprocessing through normalization method is presented. The performance of this pretreatment technique on the basic *k*-means clustering method was evaluated using actual malaria secondary data consisting of six variables and fourteen sample size selected from Katsina State Health Services Management Board (KTSHSMB), The six variables are uncomplicated malaria, clinical malaria and severe malaria from January 2016 to February 2016 denoted by X_1 to X_6 respectively and the sample sizes are Bakori (BKR), Batagarawa (BTG), Batsari (BTR), Baure (BAU), Bindawa (BDW), Charanchi (CRC), Dandume (DDM), Danja (DNJ), Daura (DRA), Dutsi (DTS), Dutsinma (DTM), Faskari (FKR), Funtua (FTA) and Ingawa (IGW) local government areas respectively. The initial data is presented in Table 4.1 below.

Table 4.0: The initial dataset consisting of six variables and fourteen sample size

LGA	X_1	X_2	X_3	X_4	X_5	X_6
BKR	919	567	25	56	303	0
BTG	502	1153	147	1035	1530	41
BTR	229	1027	0	739	1111	26
BAU	326	443	273	780	1130	133
BDW	797	1233	7	388	160	0
CRC	835	1087	33	513	711	7
DDM	924	1167	29	1591	1743	33
DNJ	544	526	0	782	1125	2
DRA	834	860	260	1119	1556	274
DTS	1003	1582	231	878	1035	160
DTM	114	238	36	1238	1595	38
FKR	872	1989	19	876	2277	94
FTA	761	948	80	1604	1628	251
IGW	355	920	8	402	707	7

The formation of clusters and the corresponding centers are shown in Figure 4.0.

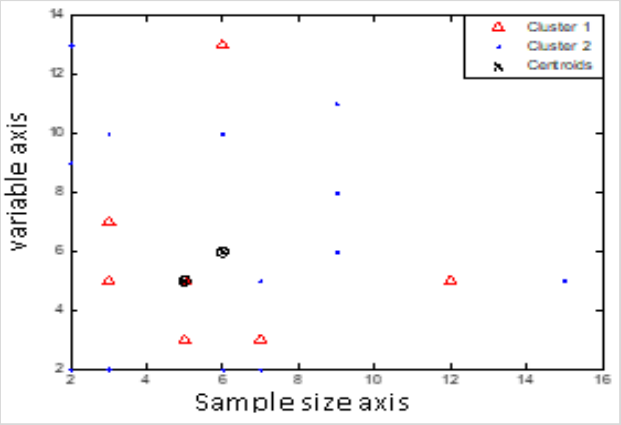

Figure 4. 0 Basic *k*-means method

Figure 4.0 shows the basic *k*-means method using the initial dataset consisting of fourteen sample size and six variables as contained in Table 4.0.

From Figure 4.0, six points lie outside the formation of the cluster, and six points lie on the boundary denoted by coordinates (2, 13), (2, 9), (2, 2), (3, 2). (6, 2) and (7, 2) are in cluster 2. This is one of the fundamental drawbacks of basic k-means. This method does not capture all variable points in forming clusters.

Normalization of data

Some tests have been done using four different normalization methods, z-score, min-max, decimal scaling, and mean absolute deviation. This is to test the performance using the basic k-means clustering algorithm and choose the appropriate data normalization method from these four data normalization methods. We measured the cluster quality among the four normalization methods using the sum-of-squares error, which represents the distance between data points and their cluster centers. The data set used for this purpose is presented in Table 4.0 above.

In the z-score, data are normalized to mean and standard deviation. The z-score normalization formula used in this research is given in equation 2 above.

Table 4.1: The normalized z-score dataset

LGA	X_1	X_2	X_3	X_4	X_5	X_6
BKR	-0.3554	0.6497	-0.6749	0.5904	0.6035	-0.8173
BTG	1.0116	1.4372	0.8677	-0.9332	1.2559	1.1058
BTR	-1.1756	-0.5316	0.0964	-0.9332	0.9297	0.1442
BAU	-0.9022	2.6184	-1.0605	1.3522	-0.7014	1.1058
BDW	0.1914	0.6497	-1.0605	0.5904	-0.3751	-0.8173
CRC	-0.3554	1.0434	1.6390	-0.9332	-0.7014	-0.8173
DDM	-0.9022	-0.1378	-0.6749	0.9713	0.9297	-0.8173
DNJ	-1.1756	-0.5316	0.0964	-0.5523	1.9083	0.1442
DRA	1.8317	-1.3191	-1.0605	-0.9332	-0.7014	1.1058
DTS	1.2850	-0.5316	-0.2892	0.2095	-1.0276	-0.8173
DTM	0.4648	-1.3191	0.8677	-0.5523	-0.3751	-0.8173
FKR	1.8317	-0.9253	-1.0605	2.1141	-1.0276	0.1442
FTA	-1.1756	-0.5316	1.2534	0.2095	-0.7014	0.1442
IGW	-0.3554	1.0434	-1.4462	2.1141	-0.7014	-0.8173

Table 4.1 gives scaled values using the z-score method containing six variables and 14 sample sizes. The formation of clusters and the corresponding centers are shown in Figure 4.1.

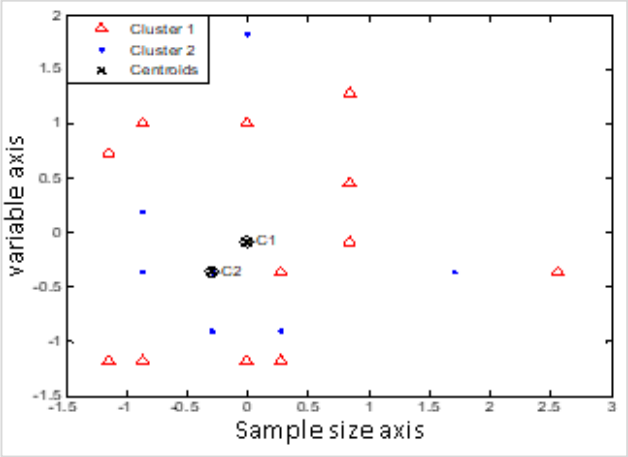

Figure 4.1 *z*-score *k*-means method

Figure 4.1 shows cluster formation obtained using normalized z-score data. It can be observed that no point is outside the cluster formation. This implies that this method is good for grouping points. However, it can be observed that most of the points are located far from the C1 and C2 centers. Moreover, the distance between the two centers C1 and C2 is very close.

Min-Max performs a linear modification of the original data set, transforming it into a rolling range from 0 to 1. i. e by subtracting the minimum value of each observation, the result is then divided by the difference between the maximum and minimum values. The formula used for min-max is given in equation 3 above.
Table 4.2 gives scaled values using the min-max method containing six variables and 14 sample sizes. The formation of clusters and the corresponding centers are shown in Figure 4.2

Table 4.2: The normalized min-max dataset

LGA	X_1	X_2	X_3	X_4	X_5	X_6
BKR	0.0050	0.0060	0.0030	0.0050	0.0060	0.0010
BTG	0.0100	0.0080	0.0070	0.0010	0.0080	0.0030
BTR	0.0020	0.0030	0.0050	0.0010	0.0070	0.0020
BAU	0.0030	0.0110	0.0020	0.0070	0.0020	0.0030
BDW	0.0070	0.0060	0.0020	0.0050	0.0030	0.0010
CRC	0.0050	0.0070	0.0090	0.0010	0.0020	0.0010
DDM	0.0030	0.0040	0.0030	0.0060	0.0070	0.0010
DNJ	0.0020	0.0030	0.0050	0.0020	0.0100	0.0020
DRA	0.0130	0.0010	0.0020	0.0010	0.0020	0.0030
DTS	0.0110	0.0030	0.0040	0.0040	0.0010	0.0010
DTM	0.0080	0.0010	0.0070	0.0020	0.0030	0.0010
FKR	0.0130	0.0020	0.0020	0.0090	0.0010	0.0020
FTA	0.0020	0.0030	0.0080	0.0040	0.0020	0.0020
IGW	0.0050	0.0070	0.0010	0.0090	0.0020	0.0010

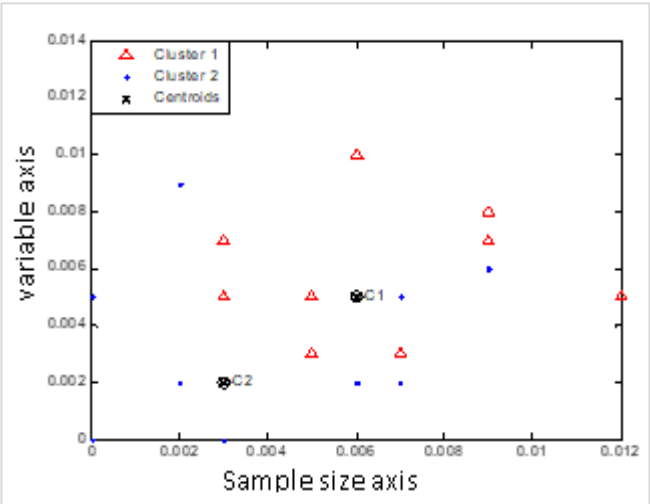

Figure 4.2 Min-Max *k*-means method

.
Figure 4.2 shows the cluster formation obtained using normalized min-max data. The corresponding data set is shown in Table 4.3. It can be observed that the three points are outside the cluster formation. These three points lie on the contour marked with the coordinates (0, 0.005), (0.003, 0) found in cluster 2 and (0.012, 0.005) in cluster 1.
Decimal scaling moves the decimal point but always keep the original numeric value. The typical scale holds values in the range [-1, 1], and the decimal point numbers are shifted by the absolute largest values in the data set. The formula used for decimal scaling is given in equation 4 above.
Table 4.3 gives the values scaled using the decimal scaling method containing six variables and 14 sample sizes. The formation of clusters and the corresponding centers are shown in Figure 4.3.

Table 4.3: The normalized decimal scaling dataset

LGA	X_1	X_2	X_3	X_4	X_5	X_6
BKR	0.2857	0.3571	0.1429	0.2857	0.3571	0.0000
BTG	0.6429	0.5000	0.4286	0.0000	0.5000	0.1429
BTR	0.0714	0.1429	0.2857	0.0000	0.4286	0.0714
BAU	0.1429	0.7143	0.0714	0.4286	0.0714	0.1429
BDW	0.4286	0.3571	0.0714	0.2857	0.1429	0.0000
CRC	0.2857	0.4286	0.5714	0.0000	0.0714	0.0000
DDM	0.1429	0.2143	0.1429	0.3571	0.4286	0.0000
DNJ	0.0714	0.1429	0.2857	0.0714	0.6429	0.0714
DRA	0.8571	0.0000	0.0714	0.0000	0.0714	0.1429
DTS	0.7143	0.1429	0.2143	0.2143	0.0000	0.0000
DTM	0.5000	0.0000	0.4286	0.0714	0.1429	0.0000
FKR	0.8571	0.0714	0.0714	0.5714	0.0000	0.0714
FTA	0.0714	0.1429	0.5000	0.2143	0.0714	0.0714
IGW	0.2857	0.4286	0.0000	0.5714	0.0714	0.0000

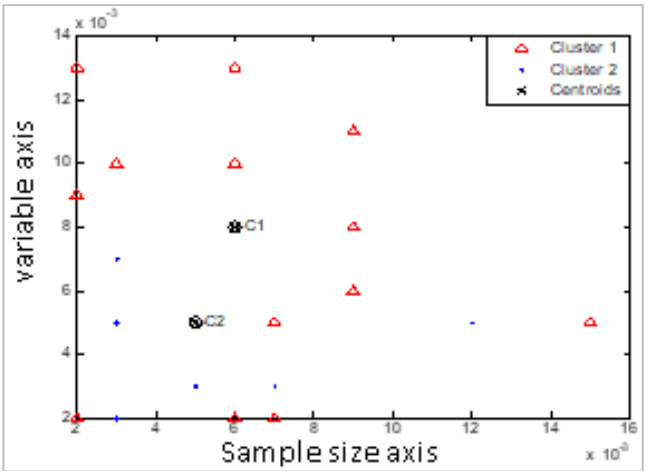

Figure 4.3 Decimal scaling *k*-means method

Figure 4.3 shows the cluster formation obtained using normalized decimal rate data. The corresponding data set is shown in Table 4.3. It can be observed that six points are outside the cluster formation. These six points lie on the boundary marked by the coordinates (0, 13), (0, .9), (2, 2), (6, .0), (7, .0) found in cluster 1 and (0 , 3) in cluster 2, where the x and y are coordinates multiplied by 10^{-3}.

The mean absolute deviation of an element of a data set is the absolute difference between this element and a given point. The formula used for mean absolute deviation is given in equation 5 above.

Table 4.4 gives scaled values using the mean absolute deviation method containing six variables and 14 sample sizes. The formation of clusters and the corresponding centers are shown in Figure 4.4.

Table 4.4: The normalized Mean Absolute deviation dataset

LGA	X_1	X_2	X_3	X_4	X_5	X_6
BKR	0.2857	0.1429	0.5000	0.2143	0.0714	0.0714
BTG	0.0000	0.4286	0.0000	0.5714	0.0714	0.0000
BTR	0.0000	0.2857	0.2143	0.1429	0.0714	0.2857
BAU	0.4286	0.2143	0.1429	0.3571	0.4286	0.0714
BDW	0.2857	0.2143	0.1429	0.3571	0.4286	0.0714
CRC	0.0000	0.1429	0.2857	0.0714	0.6429	0.5714
DDM	0.3571	0.1429	0.2143	0.2143	0.0000	0.1429
DNJ	0.0714	0.0000	0.4286	0.0714	0.1429	0.2857
DRA	0.0000	0.0714	0.0714	0.5714	0.0000	0.0714
DTS	0.2143	0.1429	0.5000	0.2143	0.0714	0.2143
DTM	0.0714	0.0714	0.0000	0.2857	0.1429	0.4286
FKR	0.5714	0.3571	0.1429	0.2857	0.3571	0.0714
FTA	0.2143	0.1429	0.4286	0.1429	0.0000	0.5000
IGW	0.5714	0.0714	0.0714	0.0000	0.2143	0.0000

Figure 4.4 shows cluster formation using normalized mean absolute deviation data. It can be observed that no point is outside the cluster formation. This implies that this method is also good for grouping points. However, it can be observed that most of the points are located far from the C1 and C2 centers. Moreover, the distance between the two centers C1 and C2 is very close. This is not the desired goal for clustering, but since all points fit into the formation, this is not the case with the other three

methods and has less total squared error and less time taken, when compared with the z-score normalization method; this makes the mean absolute deviation normalized method acceptable in this study. In this way, we infer that the mean absolute deviation is a good method for data pre-processing.

Table 4.5 summarizes clustering results from basic k-means and using four different normalization methods (z-score, min-max, decimal scaling, and mean absolute deviation). Experimental results show that the mean absolute deviation is the best among the four methods. In fact, this method finds that none of the points in the cluster are outside the cluster formation, as shown in Figure 4.4

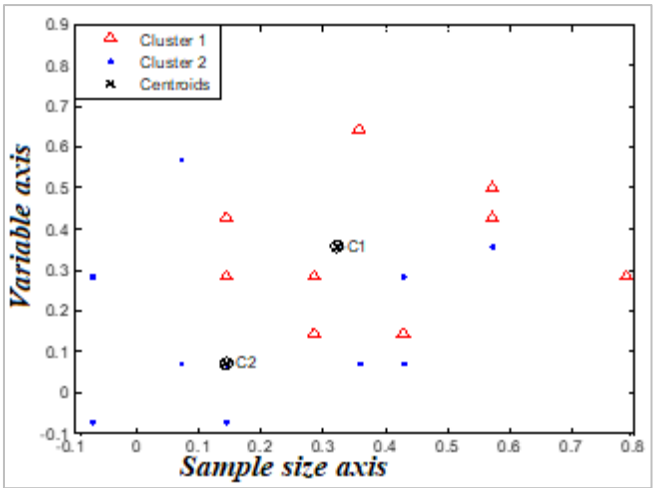

Figure 4.4: Mean absolute deviation *k*-means method

Table 4.5: Summary of cluster formation results

Method of Cluster formation	Points out of cluster formations		SSE	CPU time taken (Sec)
	cluster 1	cluster 2		
Basic *k*-means	0	6	233.08	30.00
Z-score	0	0	87.53	17.00
Min-Max	1	2	98.41	19.00
Decimal scaling	5	1	106.29	21.00
Mean absolute deviation	0	0	78.26	16.00

CONCLUSIONS

The idea behind pre-processing is to reduce the dimensionality of data that consists of a large number of variables. Therefore, the k-means results using the pre-processed data provide the expected results with improved cluster quality and detection of some clustered points. This is further supported by the sum of the squared errors and the runtime obtained by the four normalization methods compared with the basic k-means method. Therefore, based on the experimental results of this research, we can conclude that the mean absolute deviation is the best among the four normalization methods as it captures all points of clustering. Hence, the three methods of z-score, min-max and decimal scaling cannot be chosen as suitable data pre-processing methods as they cannot capture all points of clustering.

REFERENCES

Alshalabi, L., Shaaban, Z. and Kasasbeh, B. (2006). Data Mining: A Preprocessing Engine. *Journal of Computer Science,* 2(9):735-739. http://dx.doi.org/10.3844/jcssp.2006.735.739

Atomi, W.H. (2012). The effect of data preprocessing on the performance of artificial neural networks techniques for classification problems, *Doctoral dissertation, University Tun Hussein Malaysia.* https://www.semanticscholar.org/paper/The-effect-of-data-preprocessing-on-the-performance-Atomi/a218d30a0e94e72ecda2bfc63034f253bc21a79c

Chandrasekhar, T., Thangavel, K. and Elayaraja, E. (2011). Effective Clustering Algorithms for Gene Expression Data. *International Journal of Computer Applications,* 32(4): 25-29. http://research.ijcaonline.org/volume32/number4/pxc3875454.pdf

Cios, K. J., Swiniarski, R. W., Pedrycz, W. and Kurgan, L. A. (2007). Unsupervised learning: clustering in Data Mining. *Springer, Boston, MA*, 257–288. https://doi.org/10.1007/978-0-387-36795-8_9

Guojun, G., Chaoqun, M. and Jianhong, W. (2007). Data Clustering: Theory, Algorithms and Applications. *ASA-SIAM Series on Statistics and Applied Probability.* https://dl.acm.org/doi/10.5555/1296150

Hans-Joachim M., Bartel, H.G. and Dolata, J. (2008). Effects of Data Transformation on Cluster Analysis of Archaeometric Data. Proceedings of the 31st Annual Conference of the Gesellschaft für Klassifikation e.V., Albert-Ludwigs-Universität Freiburg 681-688. https://www.worldcat.org/title/5665233343

Jain, A. and Dubes, R. (1988). Algorithms for Clustering Data. Prentice-Hall. *Englewood Cliffs, NJ:* https://dl.acm.org/doi/abs/10.5555/42779

Manpreet ,K. and Usvir, K. (2013). A Survey on Clustering Principles with K-means Clustering Algorithm Using Different Methods in Detail. *International Journal of Computer Science and Mobile Computing;*

2(5):327-331. https://ijcsmc.com/docs/papers/May2013/abstracts/V2I52013120.pdf

Larose, D. T. (2005). Discovering Knowledge in Data: *An Introduction to Data Mining, New Jersey: John Wiley and Sons.* https://www.stevens.edu/sites/stevens_edu/files/CareCenter/UTC/Discovering_Knowledge_in_Data.pdf

Luai, A., Zyad, S. and Basel, K. (2006). Data Mining a Preprocessing Engine. *Journal of Computer Science.* 2(9), 735-739. http://dx.doi.org/10.3844/jcssp.2006.735.739

Milligan, G.W. and Cooper, M.C. (1988). A study of standardization of variables in cluster analysis. *Journal of Classification*, 5, 181-204. https://doi.org/10.1007/BF01897163

Manimekalai, M., Anusha, M. and Srinaganya, G. (2013). Clustering Analysis on Statistical Data Using Agglomerative Method. *International Journal of Information Sciences and Application.* 5(1): 33-38 https://www.ripublication.com/irph/ijisa/ijisav5n1_04.pdf

Karthikeyani, N, V., and Thangavel, K. (2009). Impact of Normalization in Distributed K-Means Clustering. *International Journal of Soft Computing,* 4(4): 168-172. https://medwelljournals.com/abstract/?doi=ijscomp.2009.168.172

Sairam N, Mahendiran A, Saravanan N. and Subramanian NV. (2012) Implementation of K-means clustering in cloud computing environment. *Research Journal of Applied Sciences, Engineering and Technology,* 4(10): 1391-1394 https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.458.2123

Telgarsky M, Vattani A., (2010). Method: k-means Clustering without Voronoi. *In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics,* 9: 820-827. https://proceedings.mlr.press/v9/telgarsky10a.html

Tsai, C. Y., and Chiu, C. C. (2008). Developing a Feature Weight Self-Adjustment Mechanism for a K-means Clustering Algorithm. *Computational Statistics and Data Analysis,* 52(10): 4658-4672. https://doi.org/10.1016/j.csda.2008.03.002